



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KS141501

**LAYANAN BERBASIS LOKASI UNTUK MENEMUKAN
KOKI RUMAHAN YANG TERDEKAT DAN TERSEDIA**

***LOCATION BASED SERVICE FOR FINDING NEARBY
AVAILABLE HOME COOKS***

**FAISAL SETIA PUTRA
NRP 5213 100 155**

**Dosen Pembimbing
Arif Wibisono, S.Kom., M.Sc.**

**DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017**

TUGAS AKHIR - KS141501

LAYANAN BERBASIS LOKASI UNTUK MENEMUKAN KOKI RUMAHAN YANG TERDEKAT DAN TERSEDIA

FAISAL SETIA PUTRA
NRP 5213 100 155

Dosen Pembimbing
Arif Wibisono, S.Kom., M.Sc.

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017

FINAL PROJECT - KS141501

***LOCATION BASED SERVICE FOR FINDING NEARBY
AVAILABLE HOME COOKS***

**FAISAL SETIA PUTRA
NRP 5213 100 155**

**Academic Supervisor
Arif Wibisono, S.Kom., M.Sc.**

**Information Systems Department
Faculty of Information Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2017**

LEMBAR PENGESAHAN

LAYANAN BERBASIS LOKASI UNTUK MENEMUKAN KOKI RUMAHAN YANG TERDEKAT DAN TERSEDIA

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

FAISAL SETIA PUTRA
NRP. 5213100155

Surabaya, 11 Januari 2017

**KETUA
JURUSAN SISTEM INFORMASI**

Dr. Ir. Aris Cahyanto, M.Kom
NRP. 196503101991021001

LEMBAR PERSETUJUAN

LAYANAN BERBASIS LOKASI UNTUK MENEMUKAN KOKI RUMAHAN YANG TERDEKAT DAN TERSEDIA

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

FAISAL SETIA PUTRA

NRP. 5213100155

Disetujui Tim Penguji: Tanggal Ujian: 11 Januari 2017

Periode Wisuda: Maret 2017

Arif Wibisono, S.Kom., M.Sc.

(Pembimbing I)

Mahendrawathi ER, S.T., M.Sc., Ph.D

(Penguji I)

Rully Agus Hendrawan, S.Kom., M.Eng

(Penguji II)

LAYANAN BERBASIS LOKASI UNTUK MENEMUKAN KOKI RUMAHAN YANG TERDEKAT DAN TERSEDIA

Nama Mahasiswa : Faisal Setia Putra

NRP : 5213100155

Jurusan : Sistem Informasi FTIF-ITS

Pembimbing I : Arif Wibisono, S.Kom., M.Sc.

ABSTRAK

Makanan merupakan kebutuhan pokok manusia yang berkontribusi secara langsung dalam menjaga kesehatan. Pola konsumsi makanan yang buruk dengan olahraga yang tidak teratur berpotensi menimbulkan penyakit, baik jangka pendek maupun jangka panjang. Salah satu yang menyebabkan pola konsumsi makanan yang buruk adalah karena ketidakmampuan individu dalam mengolah bahan mentah menjadi makanan jadi. Akibatnya banyak individu memilih untuk membeli makanan di luar yang banyak mengandung bahan berbahaya (misalnya MSG atau Aspartam). Di lain sisi, terdapat anggota masyarakat yang dapat mengolah bahan mentah menjadi makanan (kita sebut sebagai koki rumahan). Biasanya para koki rumahan adalah individu biasa yang tidak mendapatkan pelatihan atau pendidikan khusus dibidang kuliner. Namun, tidak menutup kemungkinan terdapat individu yang mendapatkan pelatihan dan pendidikan khusus kuliner untuk bergabung.

Tugas akhir ini bertujuan untuk mempertemukan antara individu yang tidak bisa memasak dengan koki rumahan. Individu akan menyediakan bahan baku dan tempat bagi koki rumahan. Sebagai gantinya, koki rumahan akan menyediakan

jasa mengolah bahan mentah menjadi makanan jadi. Dengan demikian, individu mendapatkan makanan secara reguler sekaligus menjaga kualitas asupan makanan karena kendali bahan baku ada di individu. Adapun disiplin ilmu yang digunakan pada penelitian ini adalah enterprise resource planning, khususnya pada proses order to cash. Proses ini dimulai ketika individu mencari koki rumahan untuk memasak makanan, sehingga terbentuk pesanan yang menghasilkan tagihan akhir.

Oleh karena itu, penulis mengajukan solusi dalam bentuk sebuah aplikasi pencari koki rumahan. Aplikasi pencari koki rumahan adalah sebuah platform berbasis web yang terintegrasi dengan Google Maps yang dapat membantu dalam mencari koki rumahan di daerah sekitar yang dapat membuatkan masakan sesuai dengan keinginan dan juga menyehatkan. Sehingga permasalahan terhadap makanan yang mengandung bahan-bahan berbahaya dapat segera teratasi.

Kata kunci : Google Maps, Location Based Service, Makanan Sehat, Personal Chef

LOCATION BASED SERVICE FOR FINDING NEARBY AVAILABLE HOME COOKS

Student Name : Faisal Setia Putra
NRP : 5213100155
Department : Sistem Informasi FTIF-ITS
Supervisor I : Arif Wibisono, S.Kom., M.Sc.

ABSTRACT

Food is one of human basic needs that directly contributed to maintain body health. The habit of consuming low quality food combined irregular exercise patterns might bring out diseases, either for short or long term. One thing that cause people regularly consuming low quality foods is the inability of someone to process raw ingredients to prepared dishes. Because of that, people preferred to buy fast foods that contains a lot of aditive materials that if consumed regularly will likely affect body health, such as MSG or Aspartam. In the other side, there are people that could process raw ingredients to prepared dishes (we call them home cooks). Usually, home cooks are untrained people that never got any specific training or education in culinary field. But, there are still possibilities that there will be people that got trained specifically in culinary to collaborate.

This thesis aims to meet individuals that cannot cook with the home cooks. The individuals will prepare the raw ingredients and a place to cook for the home cooks. In return, the home cooks will provide the service to process raw ingredients to prepared dishes. Therefore, the individuals will get prepared foods regularly and maintain the quality of their consumed foods, because the one that in charge of the raw materials is the

person himself. As for the main knowledge that used in this research is enterprise resource planning, especially in order to cash process. This process starts when a person is looking for home cooks to cook dishes, and this will form booking process that results final invoice.

Hence, the author proposed solution in the form of application for searching home cooks. Application for searching home cooks is a web-based platform that integrated with Google Maps that could help to look for home cooks in surrounding areas that could make healthy dishes according customers' needs. So, the problems regarding to foods that contains dangerous ingredients could be solved.

Keywords : Google Maps, Location Based Service, Healthy Foods, Personal Chef

KATA PENGANTAR

Puji syukur kami panjatkan kehadirat Tuhan Yang Maha Esa, karena dengan limpahan rahmat-Nya penulis dapat menyelesaikan laporan Tugas Akhir yaitu ***Layanan Berbasis Lokasi untuk Menemukan Koki Rumahan yang Terdekat dan Tersedia***. Tak lupa penulis juga mengucapkan terima kasih sebanyak-banyaknya kepada semua pihak yang sudah berkenan membantu. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada:

1. Allah SWT yang senantiasa melimpahkan berkah dan rahmat-Nya selama penulis mengerjakan Tugas Akhir.
2. Alm. Ibunda tercinta yang telah mengajarkan hal berharga dan motivasi penulis untuk selalu menjadi manusia yang bermanfaat bagi sesama.
3. Keluarga penulis, khususnya Ayah, Ibu, Kakak tercinta serta Kakek dan Nenek yang senantiasa mendoakan kelancaran serta kesuksesan dalam pengerjaan Tugas Akhir.
4. Bapak Ir. Aris Tjahyanto, M.Kom. selaku Ketua Jurusan Sistem Informasi ITS Surabaya.
5. Bapak Arif Wibisono, S.Kom., M.Sc selaku dosen wali dan juga dosen pembimbing dari Faisal Setia Putra yang memberikan motivasi sehingga saya terus berusaha mengusahakan yang terbaik selama mengerjakan Tugas Akhir.
6. Bapak Rully Agus Hendrawan, S.Kom., M.Eng selaku Ketua Laboratorium Sistem Enterprise yang bersedia memberikan fasilitas tempat untuk mengerjakan Tugas Akhir ini.

7. Responden yang dengan sukarela membantu memberikan feedback dan seluruh masukkan sehingga penulis dapat melakukan pengembangan aplikasi.
8. Teman-teman Laboratorium ADDI yang telah mempersilakan penulis bernaung dan mencari inspirasi dalam mengerjakan Tugas Akhir ini.
9. Seluruh teman – teman, Keluarga Beltranis yang selalu memberikan banyak dukungan, pengalaman, cerita, dan kenangan selama penulis melakukan penelitian.
10. Mas dan Mbak serta adik – adik jurusan Sistem Informasi yang telah memberikan semangat serta bantuan kepada penulis.
11. Seluruh civitas akademika Jurusan Sistem Informasi ITS atas dukungan yang diberikan selama pengerjaan Tugas Akhir.
12. Dan seluruh pihak yang telah membantu penulis baik secara langsung maupun tidak langsung dan telah memberikan dukungan sehingga Tugas Akhir ini dapat terselesaikan dengan baik.

Penulis menyadari bahwa laporan ini masih jauh dari kesempurnaan karena kesempurnaan sejatinya hanya milik Allah SWT, maka saran dan kritik yang konstruktif dari semua pihak sangat diharapkan demi perbaikan selanjutnya.

Surabaya, 11 Januari 2016

Penulis

DAFTAR ISI

Lembar Pengesahan	ii
Lembar Persetujuan	iii
Abstrak	iii
Abstract	v
Kata Pengantar.....	vii
Daftar Isi.....	ix
Daftar Gambar	xii
Daftar Tabel.....	xv
Daftar Bagan.....	xvii
Bab I Pendahuluan	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penulisan	3
1.6 Relevansi.....	3
Bab II Tinjauan Pustaka	5
2.1 Studi Sebelumnya	5
2.2 Landasan Teori.....	6
2.2.1 Outsourcing	6
2.2.2 Prototyping Model	7
2.2.3 Food Sharing	8
2.2.4 Order to Invoice	9
2.2.5 Sistem Layanan Berbasis Lokasi	10
2.2.6 Google Maps API	10
2.3 Kajian terhadap Solusi Serupa	10
Bab III Metodologi Penelitian	13
3.1 Tahapan pelaksanaan tugas akhir	13
3.1.1 Studi Literatur.....	14
3.1.2 Identifikasi Kebutuhan Perangkat Lunak	14
3.1.3 Analisis dan User Story	14
3.1.4 Pengkodean Aplikasi.....	14
3.1.5 Pengujian dan evaluasi	15
3.1.6 Penyusunan laporan tugas akhir	15

Bab IV Analisis dan Perancangan	17
4.1 Analisis Kebutuhan Sistem	17
4.1.1 Kebutuhan Fungsional Sistem	21
4.1.2 Kebutuhan Non-Fungsional Sistem	22
4.1.3 Penentuan Aktor	22
4.1.4 Daftar Use Case	23
4.2 Perancangan Aplikasi	24
4.2.1 Use Case Diagram	24
4.2.2 Diagram Aktivitas	26
4.3 Analisis Kebutuhan Iterasi 2	32
4.3.1 Kebutuhan Fungsional Sistem Iterasi 2	33
4.3.2 Daftar Use Case Iterasi 2	34
4.4 Perancangan Aplikasi Iterasi 2	34
4.4.1 Use Case Diagram Iterasi 2	34
4.4.2 Diagram Aktivitas Iterasi 2	35
4.5 Protokol Pengujian Aplikasi	38
Bab V Implementasi	41
5.1 Lingkungan Implementasi	41
5.1.1 Implementasi Hardware	41
5.1.2 Implementasi Software	41
5.2 Pembuatan Database	42
5.3 Implementasi Fungsi	42
5.3.1 Implementasi Login	43
5.3.2 Implementasi Pendaftaran Pelanggan	44
5.3.3 Implementasi Pendaftaran Koki Rumahan	46
5.3.4 Implementasi Kemampuan Memasak	49
5.3.5 Implementasi Pemesanan Koki	51
5.3.6 Implementasi Informasi Tagihan	53
5.3.7 Implementasi Koordinat Lokasi	54
5.3.8 Implementasi Pencarian Koki Terdekat	56
5.4 Struktur Direktori	58
Bab VI Hasil Pengujian dan Pembahasan	59
6.1 Usability Testing	59
6.1.1 Usability Testing I	59
6.1.2 Usability Testing II	60
Bab VII Kesimpulan dan Saran	63
7.1 Kesimpulan	63

7.2 Saran	64
Daftar Pustaka	65
Biodata Penulis	67
Lampiran A Usability Test Plan Dashboard.....	69
Lampiran B Hasil Translasi Wawancara.....	71
Lampiran C Hasil Perhitungan System Usability Scale (SUS)	73
Lampiran D Kode Pembuatan Aplikasi	75
Lampiran E Lembar Persetujuan	83
Lampiran F System Usability Scale Form.....	85
Lampiran G Skenario Test Case	87

DAFTAR GAMBAR

Gambar 1.1 Relevansi Laboratorium Sistem Enterprise	4
Gambar 2.1 Model Prototyping	7
Gambar 2.2 Proses Order to Invoice	9
Gambar 4.1 Use Case Diagram Aktor Koki Rumahan	25
Gambar 4.2 Use Case Diagram Aktor Pelanggan.....	26
Gambar 4.3 AD01 - Registrasi Diri.....	28
Gambar 4.4 AD02 - Tambah Kemampuan Memasak.....	29
Gambar 4.5 AD05 - Pemesanan Jasa Koki Rumahan.....	30
Gambar 4.6 AD07 - Melihat Daftar Koki Rumahan.....	31
Gambar 4.7 AD12 - Tagihan Transaksi	32
Gambar 4.8 Use Case Diagram Aktor Pelanggan Iterasi 2	35
Gambar 4.9 AD12 - Tagihan Transaksi Iterasi 2.....	36
Gambar 4.10 AD13 - Memberikan Rating dan Komentar	37
Gambar 5.1 Desain Database Aplikasi Pencarian Koki	42
Gambar 5.2 Code Login Aplikasi	43
Gambar 5.3 Tampilan Halaman Login	44
Gambar 5.4 Code Pendaftaran Pelanggan.....	45
Gambar 5.5 Tampilan Halaman Pendaftaran Pelanggan	46
Gambar 5.6 Code Pendaftaran Koki Rumahan	47
Gambar 5.7 Tampilan Halaman Pendaftaran Koki Rumahan	48
Gambar 5.8 Code Mengubah Kemampuan Memasak Koki...	49
Gambar 5.9 Tampilan Halaman Mengubah Kemampuan Memasak	50
Gambar 5.10 Code Pemesanan Koki	51
Gambar 5.11 Tampilan Halaman Pemesanan Koki Rumahan	52
Gambar 5.12 Code Informasi Tagihan.....	53
Gambar 5.13 Tampilan Halaman Informasi Tagihan	54
Gambar 5.14 Code Koordinat Lokasi	55
Gambar 5.15 Tampilan Halaman Koordinat Lokasi.....	56

Gambar 5.16 Code Pencarian Koki Terdekat	56
Gambar 5.17 Tampilan Rekomendasi Koki Berdasarkan Jarak	57
Gambar 5.18 Struktur Direktori Aplikasi Iterasi Ke-1	58
Gambar 5.19 Struktur Direktori Aplikasi Iterasi Ke-2	58
Gambar A-1 Usability Test Plan Dashboard	69
Gambar D-1 Verifikasi Pemesanan	75
Gambar D-2 Mengubah Kemampuan Koki Rumahan	75
Gambar D-3 Mengubah Informasi Pribadi	76
Gambar D-4 Mengubah Pengaturan Informasi	76
Gambar D-5 Form Pendaftaran Pelanggan	77
Gambar D-6 Form Login Aplikasi	77
Gambar D-7 Pencarian Koki Berdasarkan Kemampuan	78
Gambar D-8 Registrasi Koki Rumahan	78
Gambar D-9 Form Pemberian Rating dan Komentar	79
Gambar D-10 Verifikasi Awal Pemesanan	79
Gambar D-11 Form Verifikasi Awal Pemesanan	80
Gambar D-12 Notifikasi Pemesanan	80
Gambar D-13 Rating Koki	81
Gambar D-14 Form Tagihan	81
Gambar D-15 Riwayat Pemesanan	82
Gambar D-16 Simpan Akun	82

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 2.1 Kajian terhadap Solusi Serupa	11
Tabel 4.1 Kebutuhan Pengguna	17
Tabel 4.2 Analisis User Story	19
Tabel 4.3 Analisis Prioritas dan Kebutuhan	20
Tabel 4.4 Kebutuhan Fungsional	21
Tabel 4.5 Kebutuhan Non-Fungsional	22
Tabel 4.6 Penentuan Aktor	22
Tabel 4.7 Daftar Use Case.....	23
Tabel 4.8 Daftar Activity Diagram	27
Tabel 4.9 Perubahan Kebutuhan Fungsional Iterasi 2.....	33
Tabel 4.10 Daftar Perubahan Use Case Iterasi 2	34
Tabel 4.11 Protokol Pengujian Aplikasi	38
Tabel 5.1 Spesifikasi Hardware Server	41
Tabel 6.1 Daftar Test Case	60
Tabel 6.2 Hasil Justifikasi System Usability Scale.....	61
Tabel C-1 Hasil Perhitungan SUS	73

Halaman ini sengaja dikosongkan

DAFTAR BAGAN

Bagan 3.1 Metodologi Penelitian.....	13
--------------------------------------	----

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

Pada bab ini berisi hal-hal yang melatarbelakangi dilakukannya penelitian, rumusan dan batasan permasalahan yang dikerjakan dalam penelitian. Pada bab ini juga dijelaskan tujuan dan manfaat dilakukannya penelitian terhadap perkembangan solusi dari permasalahan yang diangkat serta metodologi dan sistematika penulisan yang digunakan dalam pelaporan tugas akhir ini.

1.1 Latar Belakang

Makanan adalah kebutuhan yang sangat penting bagi manusia karena merupakan kebutuhan pokok dan menentukan kelangsungan hidup. Kesehatan merupakan sesuatu yang penting bagi manusia, makanan yang sehat merupakan salah satu faktor yang untuk memenuhi kebutuhan kesehatan tubuh. Makanan yang sehat adalah makanan yang mengandung unsur-unsur zat yang dibutuhkan tubuh dan tidak mengandung racun. Setiap makanan selalu mengalami proses penyediaan, pemilihan bahan mentah, pengolahan, penyimpanan, pengangkutan, hingga penyajian. Dari semua tahapan tersebut memiliki risiko penyebab terjadinya keracunan pangan apabila tidak dilakukan pengawasan pangan secara baik dan benar [1]. Berdasarkan laporan BPOM selama periode tahun 2009-2013 terjadi sekitar 10.700 kasus kejadian luar biasa (KLB) keracunan pangan dengan 411.500 orang sakit dan 2.500 orang meninggal dunia [2]. Keracunan pangan ini merupakan salah satu akibat dari pola konsumsi makanan yang buruk dan olahraga yang tidak teratur. Salah satu yang menyebabkan pola konsumsi makanan yang buruk adalah karena ketidakmampuan individu dalam mengolah bahan mentah menjadi makanan jadi. Akibatnya banyak individu memilih untuk membeli makanan di luar yang banyak mengandung bahan berbahaya (misalnya MSG atau Aspartam).

Di lain sisi, terdapat anggota masyarakat yang dapat mengolah bahan mentah menjadi makanan (kita sebut sebagai koki rumahan). Koki rumahan adalah semua orang yang bisa memasak makanan rumahan dan bersedia memasak bagi orang lain. Pada bagian ini, koki rumahan berbeda dengan koki profesional yang mendapatkan pendidikan dan keterampilan khusus di bidang kuliner. Anggota masyarakat ini tidak terikat pekerjaan dengan pihak manapun, sehingga memiliki kebebasan dalam mengatur jadwal ekonominya. Biasanya para koki rumahan adalah individu biasa yang tidak mendapatkan pelatihan atau pendidikan khusus di bidang kuliner. Namun, tidak menutup kemungkinan terdapat individu yang mendapatkan pelatihan dan pendidikan khusus kuliner untuk bergabung.

Tugas akhir ini bertujuan untuk mempertemukan antara individu yang tidak bisa memasak dengan koki rumahan. Individu akan menyediakan bahan baku dan tempat bagi koki rumahan. Sebagai gantinya, koki rumahan akan menyediakan jasa mengolah bahan mentah menjadi makanan jadi. Dengan demikian, individu mendapatkan makanan secara reguler sekaligus menjaga kualitas asupan makanan karena kendali bahan baku ada di individu. Oleh karena itu, penulis mengajukan solusi dalam bentuk sebuah aplikasi pencari koki rumahan. Aplikasi pencari koki rumahan adalah sebuah platform berbasis web yang terintegrasi dengan *Google Maps* yang dapat membantu dalam mencari koki rumahan di daerah sekitar yang dapat membuatkan masakan rumahan sesuai dengan keinginan dan juga menyehatkan.

1.2 Rumusan Masalah

Permasalahan yang akan diangkat dalam tugas akhir ini adalah bagaimana menghubungkan antara individu dengan koki rumahan dengan mempertimbangkan lokasi geografis.

1.3 Batasan Masalah

Adapun batasan masalah pada tugas akhir ini adalah:

1. Aplikasi yang dikembangkan hanya dapat diakses ketika terhubung dengan jaringan internet.
2. Website dibangun dengan menggunakan PHP dan Database MySQL.
3. Koki rumahan dan individu memiliki kontak telepon yang dapat dihubungi secara nyata.
4. Browser yang dipakai mendukung Javascript dan AJAX.
5. Pengguna aplikasi merupakan orang yang terbiasa dengan internet dan dapat menggunakan gadget.

1.4 Tujuan Penelitian

Tujuan pengerjaan tugas akhir ini adalah membuat aplikasi berbasis lokasi yang dapat menghubungkan antara koki rumahan dengan individu.

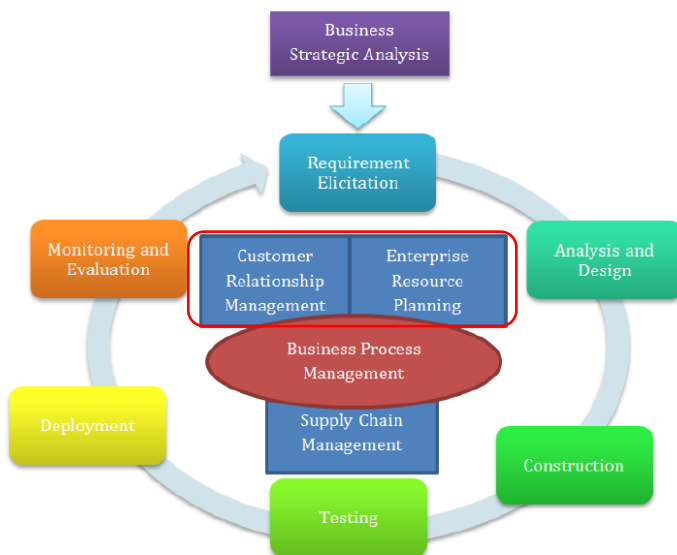
1.5 Manfaat Penulisan

Manfaat dari tugas akhir ini adalah:

1. Membantu individu untuk mengakses makanan sehat.
2. Membuka lapangan kerja bagi koki rumahan.

1.6 Relevansi

Tugas akhir ini relevan dengan standar tugas akhir untuk jenjang sarjana sistem informasi. Dalam tugas akhir ini mengembangkan sistem pencarian koki rumahan dengan memanfaatkan layanan berbasis lokasi geografis dalam bentuk web aplikasi. Manfaat aplikasi ini bermanfaat untuk individu dalam mengakses makanan sehat selain itu juga dapat membuka lapangan kerja bagi koki rumahan. Tugas akhir ini relevan dengan laboratorium sistem enterprise karena mencakup topik yang terdapat pada laboratorium sistem enterprise yaitu *Enterprise Resource Planning* dan *Customer Relationship Management*. Adapun relevansi tugas akhir ini dengan topik yang terdapat pada laboratorium sistem enterprises dapat dilihat pada Gambar 1.1



Gambar 1.1 Relevansi Laboratorium Sistem Enterprise

BAB II

TINJAUAN PUSTAKA

Bab tinjauan pustaka ini akan menjelaskan studi sebelumnya dari penelitian ini dan dasar teori dari penelitian ini.

2.1 Studi Sebelumnya

Penelitian-penelitian yang memiliki topik yang sama dengan penelitian ini (*location based service*) telah banyak dilakukan, hasil dari penelitian tersebut berupa sebuah aplikasi pencarian taksi terdekat dengan pelanggan ataupun aplikasi pemandu pencarian masjid terdekat.

Sebuah penelitian dilakukan untuk mengatasi pemesanan taksi yang selama ini masih dilakukan secara manual yaitu melalui telepon operator taksi, SMS maupun memanggil taksi secara langsung di jalan. Pada umumnya penyedia layanan taksi masih menggunakan pemancar radio untuk menginformasikan adanya pemesan jasa taksi kepada para sopir taksi. Hal tersebut masih dilakukan secara manual oleh operator layanan taksi dan belum adanya konfirmasi taksi mana yang akan melayani pemesan taksi nantinya. Pencarian taksi terdekat untuk pemesan taksi juga tidak bisa dilakukan tanpa mengetahui lokasi pemesan taksi secara otomatis. Dengan memanfaatkan fasilitas GPS, maka dikembangkan sistem untuk otomatisasi pencarian taksi terdekat untuk pemesanan taksi [3]. Pada penelitian ini, sistem yang dapat mengotomasi pencarian taksi dengan jarak terdekat terhadap posisi pemesan taksi serta mengotomasi penyebaran informasi ke taksi dan konfirmasi pemesanan layanan taksi ke pengguna layanan taksi.

Penelitian lainnya juga telah dilakukan mengenai penggunaan layanan berbasis lokasi, yaitu pada penelitian dengan judul sistem pemandu pencarian masjid terdekat berbasis lokasi di atas platform android. Pada penelitian ini menghasilkan sebuah aplikasi yang bertujuan untuk memandu dalam pencarian

masjid terdekat. Selain mencari masjid terdekat, aplikasi ini juga dapat memandu rute untuk menuju masjid tersebut [4].

Selain itu terdapat juga penelitian lainnya yang serupa yang menggunakan layanan berbasis lokasi untuk mencari rumah sakit terdekat di kota Yogyakarta. Fitur-fitur pada aplikasi ini diantaranya adalah menampilkan rumah sakit terdekat dari posisi pengguna, mencari nama dokter dan menampilkan jadwal dokter tersebut, mencari spesialis dan menampilkan jadwal dokter spesialis dan menampilkan rute dari posisi pengguna ke rumah sakit tertentu [5].

2.2 Landasan Teori

Landasan teori berisi mengenai dasar pengetahuan yang digunakan dalam pengembangan tugas akhir ini.

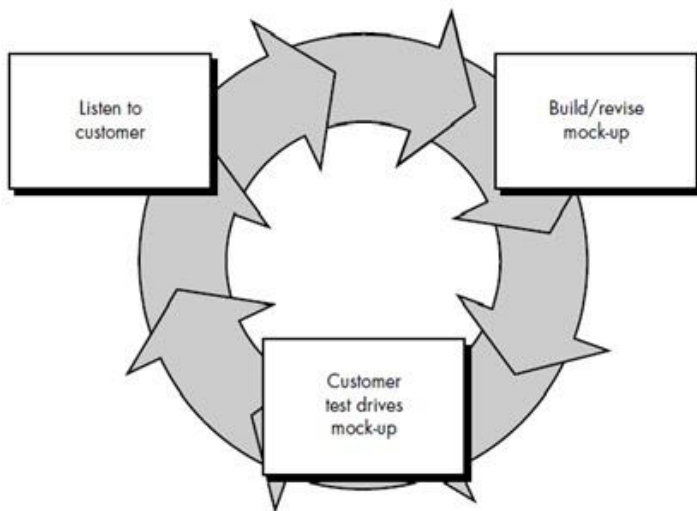
2.2.1 *Outsourcing*

Outsourcing merupakan bahasa asing yang berasal dari dua suku kata *Out* yang berarti “luar” dan *Source* yang artinya “sumber”. Namun jika diterjemahkan ke dalam Bahasa Indonesia, *Outsourcing* adalah “alih daya” [6]. Pengaturan mengenai *outsourcing* dapat dilihat dalam ketentuan Pasal 64 Undang-Undang Nomor 13 Tahun 2003 tentang Ketenagakerjaan yang menyatakan tentang suatu perjanjian kerja yang dibuat antara pengusaha dengan tenaga kerja, dimana perusahaan tersebut dapat menyerahkan sebagian pelaksanaan pekerjaan kepada perusahaan lainnya melalui perjanjian pemborongan pekerjaan yang dibuat secara tertulis [7]. Secara umum, pegawai *outsourcing* dibagi menjadi dua jenis yaitu pegawai *outsourcing* borongan dan pegawai *outsourcing* perorangan. Pegawai *outsourcing* borongan merupakan karyawan atau tenaga kerja yang bekerja pada sebuah perusahaan penyedia jasa *outsourcing* untuk mengerjakan sebuah pekerjaan khusus tertentu dan dalam waktu tertentu. Sedangkan pegawai *outsourcing* perorangan merupakan tenaga kerja yang bekerja pada sebuah perusahaan secara langsung untuk mengerjakan suatu pekerjaan khusus dengan tenggang waktu tertentu. Tenaga kerja memiliki beberapa macam jenis

diantaranya adalah tenaga kerja rohani dan tenaga kerja jasmani. Tenaga kerja jasmani terbagi menjadi tiga jenis, salah satunya adalah tenaga kerja tak terdidik. Tenaga kerja tak terdidik adalah tenaga kerja yang tidak memerlukan pelatihan ataupun pendidikan khusus.

2.2.2 *Prototyping Model*

Prototype model adalah salah satu metode pengembangan perangkat lunak yang banyak digunakan. Dengan menggunakan metode *prototyping* ini, pengembangan dan pelanggan dapat saling berinteraksi selama proses pembuatan sistem. Seringkali seorang pelanggan hanya mendefinisikan secara umum apa saja yang dibutuhkan tanpa detail input, proses, maupun output. Pada sisi yang lain, tim pengembang (*developer*) tidak yakin terhadap efisiensi dari algoritma yang digunakan, tingkat adaptasi terhadap sistem operasi ataupun rancangan tampilan antar muka [8]. Ketika situasi seperti ini, model *prototyping* sangat membantu dalam proses pembangunan sistem.



Gambar 2.1 Model *Prototyping*

Adapun tahapan-tahapan dalam pengembangan dengan menggunakan model *Prototype* yaitu sebagai berikut:

1. Mendengarkan Pelanggan
Pada tahap ini dilakukan pengumpulan kebutuhan dari sistem dengan cara mendengar keluhan dari pelanggan. Dalam membuat suatu sistem yang sesuai dengan kebutuhan, maka harus diketahui terlebih dahulu bagaimana kondisi sistem yang sedang berjalan saat ini untuk kemudian mengetahui masalah yang terjadi [8].
2. Merancang dan Membuat *Prototype*
Pada tahapan ini, dilakukan perancangan dan pembuatan *prototype* dari sistem. *Prototype* yang dibuat disesuaikan dengan kebutuhan sistem yang telah didefinisikan sebelumnya dari keluhan pelanggan maupun pengguna [8].
3. Uji Coba
Setelah *prototype* jadi, maka dilakukan uji coba oleh pelanggan maupun pengguna. Kemudian dilakukan evaluasi kekurangan-kekurangan maupun masukan dari kebutuhan pelanggan. Lalu dilakukan pengembangan kembali dengan melalui tahapan dari awal yaitu mendengarkan keluhan dari pelanggan untuk memperbaiki *prototype* yang telah ada [8].

2.2.3 *Food Sharing*

Layanan untuk berbagi makanan (*food sharing*) untuk membantu meningkatkan ekonomi sekitar sudah banyak dilakukan di negara-negara maju, seperti Inggris dan Amerika [9]. Layanan berbagi makanan ini juga berjalan di daerah Akulivik, dimana layanan berbagi makanan yang dimaksud adalah sistem barter antara pemburu dengan penduduk sekitar. Pemburu menukar hasil buruannya dengan sumber makanan yang digunakan untuk melakukan perburuan [10]. Konsep *food sharing* juga sudah digalakkan untuk membantu warga kota yang mengalami kemiskinan dan tidak memiliki tempat tinggal. Berbagi makanan dikhususkan di tempat-tempat umum yang ramai, dimana warga kota yang membutuhkan tidak susah untuk mencari makanan untuk kelangsungan hidupnya, dan

menjaga kebutuhan nutrisi warga kota agar tidak mengalami penurunan nutrisi sehingga kehidupan di kota atau daerah tersebut tidak mengalami krisis [11].

2.2.4 *Order to Invoice*

Order to Invoice adalah proses yang merupakan bagian dari *order to cash*. *Order to Cash* (OTC) adalah sekumpulan dari proses bisnis yang didalamnya terdapat penerimaan dan pemenuhan permintaan terhadap barang ataupun jasa dari pelanggan. Berbeda dengan *order to cash* yang memiliki tahapan proses hingga *payment*, *order to invoice* hanya memiliki tahapan proses hingga *invoicing*.



Gambar 2.2 Proses *Order to Invoice*

Terdapat beberapa tahapan pada *Order to Invoice* yaitu [12]:

1. *Customer Order*
Merupakan sebuah proses penerimaan pemesanan barang atau jasa.
2. *Order Fulfillment*
Pemenuhan pemesanan pelanggan terhadap barang atau jasa.
3. *Invoicing*
Proses pembuatan tagihan yang akan diberikan kepada pelanggan yang memesan barang atau jasa.

2.2.5 Sistem Layanan Berbasis Lokasi

Sistem layanan berbasis lokasi atau yang biasa dikenal dengan *Location-based Services* (LBS), menggabungkan antara proses dari layanan *mobile* dengan posisi geografis dari penggunaannya [13]. LBS mengarah pada aplikasi-aplikasi dan layanan-layanan yang menyediakan informasi lokasi mengenai pengguna *mobile*, untuk menyediakan sebuah layanan atau informasi khusus pada pengguna. Terdapat dua tipe layanan yang bisa digunakan dalam LBS untuk memperoleh posisi pengguna, yaitu dengan menggunakan posisi sel jaringan atau GPS. Dari kedua cara ini akan didapatkan posisi pengguna dalam bentuk koordinat *latitude* dan *longitude*. *Latitude* adalah representasi dari arah Utara-Selatan, sedangkan *longitude* adalah representasi dari arah Timur-Barat [14].

2.2.6 Google Maps API

Google Maps API merupakan pengembangan teknologi dari Google yang digunakan untuk menanamkan *Google Map* di suatu aplikasi yang tidak dibuat oleh Google. *Google Maps* dibuat dengan menggunakan kombinasi dari gambar peta, *database*, serta objek-objek yang interaktif yang dibuat dengan Bahasa pemrograman HTML, *Javascript* dan AJAX, dan beberapa Bahasa pemrograman lainnya [15]. *Google Maps API* adalah suatu *library* yang berbentuk *javascript* yang berguna untuk memodifikasi peta yang ada di *Google Maps* sesuai kebutuhan. Untuk membangun aplikasi yang memanfaatkan *Google Maps* di desktop dan mobile device, maka digunakan *Google Maps Javascript API v3* yang memiliki keunggulan lebih cepat dari sebelumnya [16]. *Google Maps API* ini dapat digunakan secara gratis.

2.3 Kajian terhadap Solusi Serupa

Kajian terhadap solusi serupa telah banyak dilakukan, diantaranya adalah studi yang mengembangkan perangkat pencarian ataupun pengantaran makanan untuk membantu individu dalam memenuhi kebutuhan sehari-hari. Hasil dari beberapa solusi serupa berupa aplikasi diantaranya yaitu seperti yang terlihat pada Tabel 2.1.

Tabel 2.1 Kajian terhadap Solusi Serupa

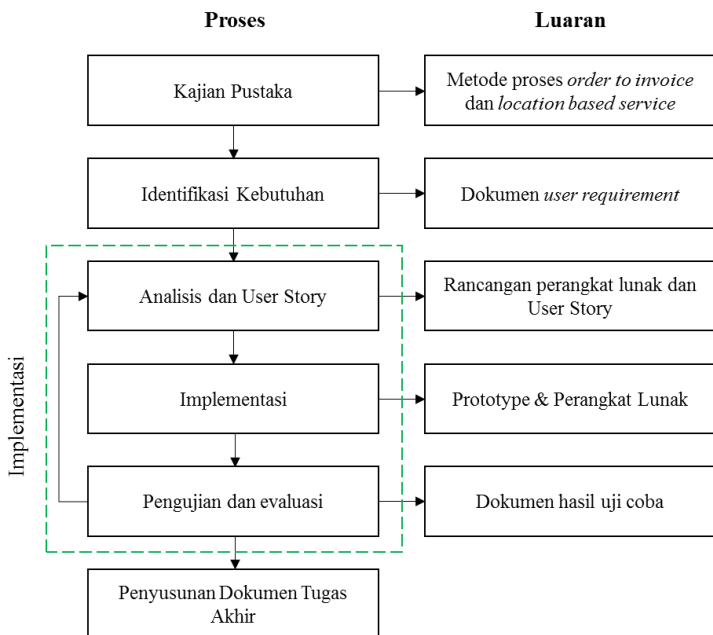
No	Nama	Penjelasan
1.	Olio	OLIO merupakan aplikasi yang menghubungkan antara pemilik makanan berlebih dengan orang-orang yang menginginkan makanan tersebut. OLIO menggunakan layanan berbasis lokasi untuk menampilkan produk-produk makanan yang dimiliki oleh pemilik makanan tersebut.
2.	Go-Food	Go-Food merupakan layanan antar pesanan makanan (<i>food delivery</i>). Dengan layanan Go-Food ini, pelanggan dapat membeli makanan di resto/rumah makan yang telah bekerjasama dengan Go-Jek.
3.	Gorry Gourmet	Gorry Gourmet merupakan sebuah aplikasi penyedia makanan sehat. Layanan ini menyediakan paket makanan diet mayo yang diklaim sudah diuji oleh ahli gizi. Selain paket diet mayo, Gorry Gourmet juga menyediakan makanan ringan sehat lainnya.

Halaman ini sengaja dikosongkan

BAB III

METODOLOGI PENELITIAN

Bab ini berisi penjelasan mengenai tahapan – tahapan yang digunakan dalam pembuatan penelitian. Setiap luaran dari proses yang dilakukan akan menjadi masukan untuk proses berikutnya. Metodologi dari penelitian ini digambarkan pada Bagan 3.1.



Bagan 3.1 Metodologi Penelitian

3.1 Tahapan pelaksanaan tugas akhir

Terdapat beberapa tahapan-tahapan pada pelaksanaan tugas akhir ini diantaranya yaitu:

3.1.1 Studi Literatur

Studi literatur merupakan upaya yang dilakukan untuk mendapatkan referensi mengenai informasi dan data yang diperlukan dalam pembuatan tugas akhir ini. Tujuan dari tahap ini adalah agar dalam penelitian ini memiliki dasar teori dan teknik yang jelas untuk digunakan dalam pemecahan masalah yang dibahas. Adapun literatur yang digunakan yaitu buku, dokumentasi, jurnal ilmiah, *e-book*, informasi dari internet dan sumber lain yang terkait. Luaran dari proses ini adalah metode proses model dari aplikasi yang akan menjadi masukan dalam analisis selanjutnya.

3.1.2 Identifikasi Kebutuhan Perangkat Lunak

Tahap ini adalah tahap identifikasi kebutuhan yang ada pada perangkat lunak. Kebutuhan perangkat lunak didapatkan dari wawancara, pemahaman studi literatur dan perbandingan dengan aplikasi yang sejenis. Pada tahapan ini seluruh kebutuhan perangkat lunak yang ada dipetakan menjadi kebutuhan fungsional dan non-fungsional ke dalam perangkat lunak berbasis web.

3.1.3 Analisis dan User Story

Pada tahap ini dilakukan desain aplikasi pencarian koki rumahan (Home Cooks) berbasis web. Kemudian, data yang didapat dari analisa kebutuhan diubah kedalam bentuk kebutuhan pengguna yang nantinya akan dibuat menjadi *user story* aplikasi. *User story* aplikasi digunakan untuk mendapatkan model proses yang efektif dan sesuai dengan kebutuhan dan permasalahan yang akan diselesaikan. Pada tugas akhir ini database yang digunakan adalah MySQL. Bahasa pemrograman yang digunakan dalam pembuatan aplikasi adalah *PHP hypertext Processor* (PHP).

3.1.4 Pengkodean Aplikasi

Pada tahap ini dilakukan pengkodean yang didasarkan pada desain aplikasi. Pada tahapan ini menggunakan metode *software development life cycle* (SDLC) menggunakan pendekatan model *prototype*. Pembuatan aplikasi web

dilakukan dengan menggunakan bahasa pemrograman *PHP* dan juga dengan menggunakan *Javascript*. Hasil dari tahapan ini berupa aplikasi yang sesuai dengan rancangan dari skenario proses.

3.1.5 Pengujian dan evaluasi

Pada tahap ini dilakukan pengujian aplikasi pencarian koki rumahan. Pengujian dilakukan untuk menguji fungsional aplikasi. Uji coba ini dilakukan untuk mendapatkan kesesuaian antara aplikasi dengan kebutuhan yang telah dirumuskan di awal pengerjaan. Pengujian dari implementasi ini akan dilakukan dalam beberapa kali iterasi. Hal ini bertujuan agar implementasi *prototype* telah benar-benar sesuai dengan kebutuhan pengguna. Calon pengguna aplikasi akan diminta untuk melakukan evaluasi seluruh fitur yang ada, memberikan kelebihan dan kelemahan dari aplikasi, apa yang perlu ditambahkan maupun dikurangi pada aplikasi. Apabila terdapat fungsi aplikasi yang kurang atau tidak bekerja sesuai kebutuhan yang ada maka harus dilakukan perbaikan pada kode pemrograman untuk menambah atau memperbaiki fungsi aplikasi agar sesuai dengan skenario proses.

3.1.6 Penyusunan laporan tugas akhir

Tahap ini merupakan tahapan terakhir dalam pengerjaan tugas akhir. Tahapan ini mendokumentasikan seluruh tahapan yang dilakukan dan seluruh luaran dari setiap proses yang dijalani. Luaran dari proses ini adalah buku Laporan tugas akhir yang disesuaikan dengan format yang sudah ditetapkan oleh Jurusan Sistem Informasi FTIF ITS Surabaya.

Halaman ini sengaja dikosongkan

BAB IV

ANALISIS DAN PERANCANGAN

Pada bab ini akan dijelaskan mengenai tahapan – tahapan yang termasuk ke dalam metode implementasi. Tahapan – tahapan tersebut meliputi mendengarkan pelanggan, membangun dan memperbaiki *prototype* hingga pengujian *prototype*. Pengembangan aplikasi dilakukan menggunakan pendekatan *Prototyping Model*, oleh karena itu, proses pengembangan akan dibagi menjadi beberapa kali iterasi. Tiap iterasi akan menggambarkan adanya perubahan kebutuhan.

4.1 Analisis Kebutuhan Sistem

Tahapan ini dimulai dengan pengumpulan kebutuhan. Pada tahap ini dilakukan analisis kebutuhan dan mengidentifikasi segala kebutuhan. Kebutuhan tersebut didapatkan dari beberapa kali wawancara dengan koki rumahan yang merupakan calon pengguna aplikasi. Oleh karena itu, dilakukan analisis terhadap kebutuhan pengguna dari aplikasi yang akan dikembangkan. Kebutuhan pengguna awal dari aplikasi didapatkan berdasarkan hasil dari pengumpulan data yang telah dilakukan. Sehingga dihasilkan kebutuhan pengguna seperti yang terdapat pada Tabel 4.1. Kebutuhan pengguna yang ada telah melalui tahapan iterasi sebanyak 2 kali dari hasil *usability testing*.

Tabel 4.1 Kebutuhan Pengguna

ID UN	Kebutuhan Pengguna
Tujuan : Membuat aplikasi yang dapat menghubungkan koki rumahan dengan individu yang membutuhkannya.	
UN001	Pengguna aplikasi dapat melihat daftar koki rumahan yang tersedia.
UN002	Pengguna aplikasi dapat dengan mudah menemukan koki rumahan yang sesuai dengan kebutuhannya.

ID UN	Kebutuhan Pengguna
UN003	Pengguna aplikasi dapat mempromosikan kemampuan memasaknya kepada orang lain.
UN004	Pengguna aplikasi dapat dengan mudah melakukan pemesanan jasa kepada koki rumahan.
UN005	Pengguna aplikasi dapat melakukan pengaturan terhadap ketersediaan jasa memasak.
UN006	Pengguna aplikasi dapat dengan mudah memantau status dari koki rumahan yang telah di pesan.
Tujuan : Menyediakan sistem yang dapat membantu mempermudah pengguna dalam memilih koki rumahan.	
UN007	Pengguna aplikasi dapat dengan mudah menemukan koki rumahan di sekitarnya.
UN008	Pengguna aplikasi dapat memberikan <i>rating</i> dan komentar terhadap koki rumahan yang telah dipesan.
UN009	Pengguna aplikasi dapat melihat <i>rating</i> dan komentar individu yang pernah melakukan pemesanan.
Tujuan : Menyediakan sistem yang dapat merekap seluruh riwayat transaksi yang pernah dilakukan.	
UN010	Pengguna aplikasi dapat dengan mudah mengakses riwayat transaksi yang pernah dilakukan.
UN011	Pengguna aplikasi dapat melihat rincian tagihan dari setiap transaksi.
UN012	Pengguna aplikasi dapat dengan mudah mendapatkan pemberitahuan informasi tagihan melalui email.

Untuk mendapatkan kebutuhan fungsional, penulis melakukan wawancara langsung kepada koki rumahan. Wawancara

dilakukan pada tanggal 20 Desember 2016 kepada pak Winarto. Pak Winarto adalah seorang koki yang kesehariannya melayani orang-orang untuk membuatkan makanan. Beliau memiliki beberapa pelanggan diantaranya adalah organisasi, instansi maupun individu. Karena kesibukan beliau yang begitu padat, ia hanya dapat dihubungi melalui nomor handphone. Selama ini, Pak Winarto menerima pesanan hanya melalui telepon ataupun bertemu langsung dengan yang bersangkutan. Seringkali, banyak tawaran pemesanan yang membutuhkan jasa Pak Winarto. Karena banyaknya tawaran, Pak Winarto bingung untuk memprioritaskan pelanggannya. Oleh karena itu, pak Winarto Butuh sebuah solusi yang dapat membuat Pak Winarto lebih mudah untuk dihubungi oleh pelanggannya dan dapat memprioritaskan pelanggannya. Dari permasalahan tersebut muncullah sebuah kebutuhan. Kemudian setelah itu aplikasi dapat dibuat berdasarkan dari kebutuhan Pak Winarto yang merupakan koki rumahan. Berikut ini merupakan kebutuhan-kebutuhan pada aplikasi berdasarkan analisis dari permasalahan diatas seperti yang terdapat pada Tabel 4.2.

Tabel 4.2 Analisis *User Story*

ID User Story	ID QA	Aktor	Cerita
US01	QA6	Koki Rumahan Pelanggan	Orang-orang yang butuh jasa saya biasanya agak kesusahan bertemu sama saya karena saya memang jarang di rumah.
US02	QA6	Koki Rumahan Pelanggan	Orang-orang yang butuh jasa saya biasanya agak kesusahan bertemu sama saya karena saya memang jarang di rumah, makanya jika ada pesanan, mereka langsung telpon saya.

ID User Story	ID QA	Aktor	Cerita
US03	QA6	Koki Rumahan Pelanggan	Kadang ada juga yang pesan ke saya tempatnya jauh tapi pembayarannya tidak termasuk biaya transport.
US04	QA6	Koki Rumahan Pelanggan	Selain itu juga kadang biaya jasa yang disepakati di awal tidak dibayar sepenuhnya.

Dari hasil analisis *user story* diatas, maka kebutuhan-kebutuhan pada aplikasi dapat diprioritaskan berdasarkan proses bisnis yang telah berjalan saat ini. Tabel prioritas berdasarkan kebutuhan dapat dilihat pada Tabel 4.3

Tabel 4.3 Analisis Prioritas dan Kebutuhan

Id User Story	Prioritas	Kebutuhan	ID User Need
US01	High	Sistem harus dapat menyediakan pendaftaran bagi koki rumahan dan pelanggan.	UN002, UN004, UN005
US02	High	Sistem harus dapat menyediakan pemesanan jasa memasak makanan koki rumahan.	UN003, UN004, UN005
US03	High	Sistem harus dapat menyediakan informasi koki rumahan berdasarkan lokasi terdekat.	UN001, UN002, UN007

ID User Story	Prioritas	Kebutuhan	ID User Need
US04	High	Sistem harus menyediakan rincian tagihan transaksi pemesanan jasa koki rumahan.	UN006, UN010, UN011

4.1.1 Kebutuhan Fungsional Sistem

Sistem yang sedang berjalan saat ini untuk mencari koki rumahan masih menggunakan cara manual yaitu didapatkan dari rekomendasi tetangga-tetangga sekitar. Selain itu, seringkali rekomendasi yang diberikan tidak sesuai dengan keinginan para pencari jasa koki rumahan. Berdasarkan hasil penggalan kebutuhan dengan aktor yang terlibat langsung dengan sistem, penulis mendapatkan kebutuhan fungsional seperti yang terlampir dalam Tabel 4.4. Kebutuhan fungsional adalah sebuah tuntutan kepada pengembang agar sistem yang dikembangkan dapat memenuhi kebutuhan dari pengguna aplikasi. Setelah mendapatkan kebutuhan fungsional, maka dibuatlah aplikasi sesuai dengan kebutuhan tersebut.

Tabel 4.4 Kebutuhan Fungsional

ID	Kebutuhan Fungsional	Referensi
KF01	Sistem menyediakan pendaftaran bagi koki rumahan dan juga pelanggan.	US01
KF02	Sistem menyediakan pemesanan jasa memasak makanan koki rumahan.	US02
KF03	Sistem menyediakan informasi koki rumahan berdasarkan lokasi terdekat.	US03

KF04	Sistem harus menyediakan rincian tagihan transaksi pemesanan jasa koki rumahan.	US04
------	---	------

4.1.2 Kebutuhan *Non-Fungsional* Sistem

Pada tahapan ini dilakukan pengelompokan kebutuhan berdasarkan area non-fungsional yang berhubungan dengan perangkat lunak. Berikut ini merupakan bagan untuk menentukan kebutuhan non-fungsional, yang dapat dilihat pada Tabel 4.5.

Tabel 4.5 Kebutuhan *Non-Fungsional*

ID	Kebutuhan Non-Fungsional
KnF01	Sistem dapat diakses 24 jam/7 hari
KnF02	Aplikasi berbasis web dapat diakses menggunakan <i>browser</i> chrome dan firefox.
KnF03	Dapat diterapkan dalam berbagai ukuran tampilan web browser.

4.1.3 Penentuan Aktor

Berdasarkan hasil dari wawancara yang telah dilakukan, dapat diketahui terdapat dua aktor penting yaitu Koki Rumahan dan Pencari Koki Rumahan. Tiap-tiap aktor tersebut memiliki kepentingan seperti yang terdapat pada Tabel 4.6.

Tabel 4.6 Penentuan Aktor

Aktor	Kepentingan
Koki Rumahan	Mengelola seluruh aktifitas aplikasi yang berhubungan dengan penawaran jasa memasak sesuai dengan keinginan konsumen.

Pelanggan	Semua entitas dalam aplikasi yang berperan sebagai konsumen layanan pencarian jasa memasak.
-----------	---

4.1.4 Daftar Use Case

Berdasarkan analisis kebutuhan yang telah dilakukan, fungsionalitas dari perangkat lunak yang dibuat dapat direpresentasikan ke dalam bentuk *use case*. *Use case* didapatkan dari kebutuhan fungsional dimana kebutuhan fungsional didapatkan dari hasil analisis *user story*. Setelah mendapatkan kebutuhan fungsional, maka dapat dipecah menjadi *use case*. Daftar *use case* berdasarkan kebutuhan fungsional perangkat lunak dapat dilihat pada Tabel 4.7 di bawah ini.

Tabel 4.7 Daftar *Use Case*

ID KF	ID Use Case	Use Case
KF01	UC01	Koki rumahan dapat mendaftarkan diri
	UC02	Koki rumahan dapat memasukkan kemampuan memasak jenis makanan.
	UC03	Koki rumahan dapat mengubah kemampuan memasak jenis makanan.
	UC04	Pelanggan dapat mendaftarkan diri.
KF02	UC05	Pelanggan dapat memesan jasa koki rumahan.
	UC06	Koki rumahan mendapatkan informasi jika terjadi transaksi.
KF03	UC07	Pelanggan dapat melihat daftar koki rumahan

	UC08	Koki rumahan dapat mengubah koordinat lokasi tempat tinggal.
	UC09	Pelanggan mendapatkan informasi koki rumahan terdekat.
KF04	UC10	Koki rumahan mendapatkan daftar riwayat pemesanan jasa.
	UC11	Koki rumahan mendapatkan informasi tagihan pelanggan.
	UC12	Pelanggan mendapatkan tagihan pada saat telah terjadi transaksi.

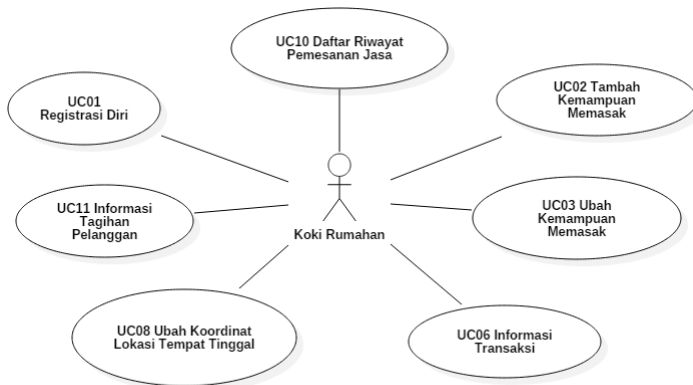
4.2 Perancangan Aplikasi

Setelah melakukan analisis terhadap kebutuhan sistem hingga menghasilkan kebutuhan fungsional dan daftar *use case*, maka tahapan selanjutnya adalah perancangan aplikasi. Pada tahapan perancangan aplikasi ini akan dijelaskan mengenai pembuatan diagram *use case* dan *activity diagram*. Aplikasi ini dibuat berbasis web agar mudah diakses oleh pengguna melalui desktop maupun smartphone. Pengguna mengakses aplikasi melalui koneksi internet dengan tampilan antar muka yang telah disiapkan oleh web server. Kemudian Web Server akan mengolah data dalam database server sesuai dengan informasi yang diminta dan mengurutkan informasi tersebut berdasarkan jarak dari lokasi pengguna aplikasi.

4.2.1 Use Case Diagram

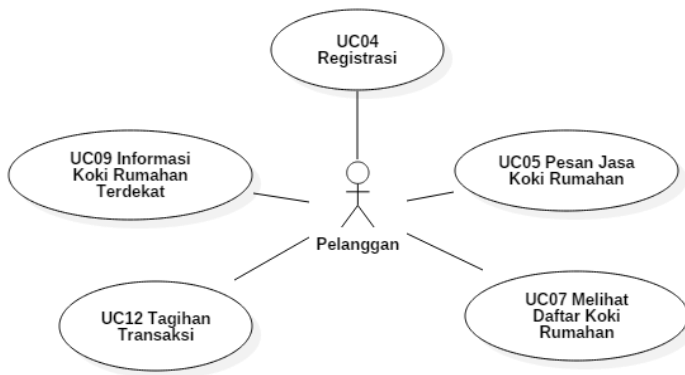
Use Case Diagram adalah diagram *use case* yang menggambarkan sekelompok *use case* dan aktor yang disertai dengan hubungan diantaranya. Diagram *use case* ini menjelaskan kebutuhan yang diinginkan oleh pengguna aplikasi. Kebutuhan-kebutuhan tersebut didapatkan dari hasil wawancara beberapa kali kepada calon pengguna aplikasi. Dari hasil kebutuhan tersebut kemudian dipetakan dan dibagi menjadi *use case* pada setiap aktor. Dimana kemudian aktor tersebut dibagi menjadi dua yaitu aktor sebagai koki rumahan

dan aktor sebagai pelanggan. Berikut ini merupakan tabel daftar *use case* yang digunakan pada saat pembuatan aplikasi.



Gambar 4.1 *Use Case* Diagram Aktor Koki Rumahan

Pada Gambar 4.1 diagram *use case* aktor koki rumahan dapat diketahui bahwa koki rumahan memiliki beberapa *use case*. Beberapa aktivitas yang dapat dilakukan oleh koki rumahan diantaranya adalah koki rumahan dapat melakukan registrasi diri. Koki rumahan melakukan pendaftaran diri untuk mempromosikan jasanya dalam memasak. Selain itu, agar pelanggan dapat mempromosikan jasanya dalam memasak, maka koki harus dapat menambahkan ataupun mengurangi daftar kemampuan memasaknya. Koki rumahan harus memasukkan koordinat lokasi tempat tinggal untuk dijadikan bahan pertimbangan bagi pelanggan dalam memilih lokasi koki rumahan yang terdekat. Setelah pelanggan melakukan proses transaksi, maka akan tampil informasi tagihan kepada pelanggan sebagai bukti pembayaran. Koki juga dapat untuk melihat transaksi apa saja yang pernah dilakukan.



Gambar 4.2 *Use Case* Diagram Aktor Pelanggan

Pada Gambar 4.2, diagram *use case* aktor pelanggan dapat diketahui bahwa pelanggan memiliki beberapa *use case*. Beberapa aktivitas yang dapat dilakukan oleh pelanggan diantaranya adalah pelanggan dapat melakukan pendaftaran diri. Pelanggan melakukan pendaftaran diri untuk menemukan koki yang cocok dengan kebutuhannya dan juga terdekat dari lokasi pelanggan. Selain mencari koki rumahan, pelanggan juga dapat memesan jasa koki tersebut untuk memasak dirumahnya. Setelah koki memasak, maka akan muncul informasi tagihan yang harus dibayarkan kepada koki rumahan tersebut. Pelanggan juga dapat untuk melihat seluruh transaksi apa saja yang pernah dilakukan.

4.2.2 Diagram Aktivitas

Diagram aktivitas digunakan untuk menggambarkan proses bisnis (alur kerja) pada aplikasi. Sebuah diagram aktivitas digunakan untuk menunjukkan suatu alur kegiatan secara berurutan. *Activity* diagram dibuat berdasarkan *use case* yang sebelumnya telah dibuat. Daftar *activity* diagram serta *mappingnya* dengan *use case* dapat dilihat pada Tabel 4.8 di bawah ini.

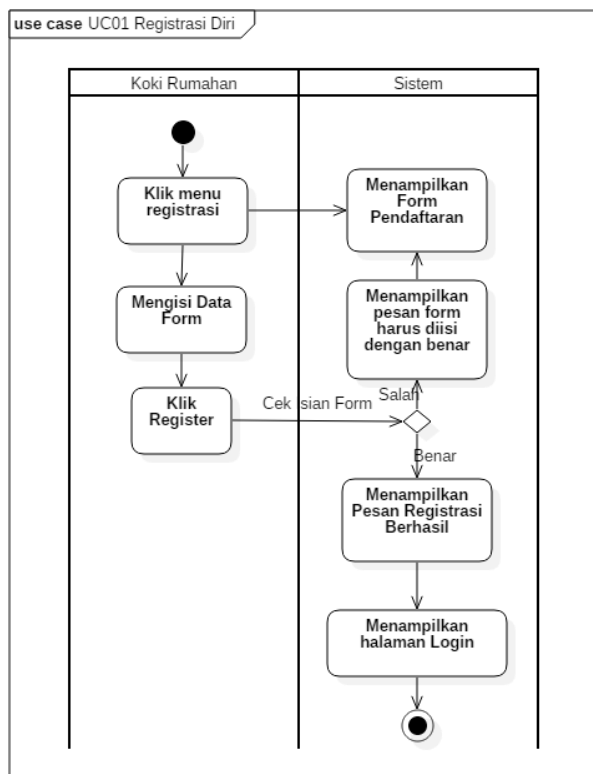
Tabel 4.8 Daftar *Activity Diagram*

ID UC	ID AD	Nama Aktivitas	Aktor
UC01	AD01	Pendaftaran diri	Koki Rumahan
UC02	AD02	Tambah Kemampuan Memasak	Koki Rumahan
UC03	AD03	Ubah Kemampuan Memasak	Koki Rumahan
UC04	AD04	Registrasi diri	Pelanggan
UC05	AD05	Pemesanan Jasa Koki Rumahan	Pelanggan
UC06	AD06	Informasi Transaksi	Koki Rumahan
UC07	AD07	Melihat Daftar Koki Rumahan	Pelanggan
UC08	AD08	Ubah Koordinat Lokasi Tempat Tinggal	Koki Rumahan
UC09	AD09	Informasi Koki Rumahan Terdekat	Pelanggan
UC10	AD10	Daftar Riwayat Pemesanan Jasa	Koki Rumahan
UC11	AD11	Informasi Tagihan Pelanggan	Koki Rumahan
UC12	AD12	Tagihan Transaksi	Pelanggan

Diagram aktivitas-aktivitas yang utama pada aplikasi ini akan dijelaskan pada Gambar 4.3, Gambar 4.4, Gambar 4.5, Gambar 4.6 dan Gambar 4.7.

4.2.2.1 Pendaftaran Diri

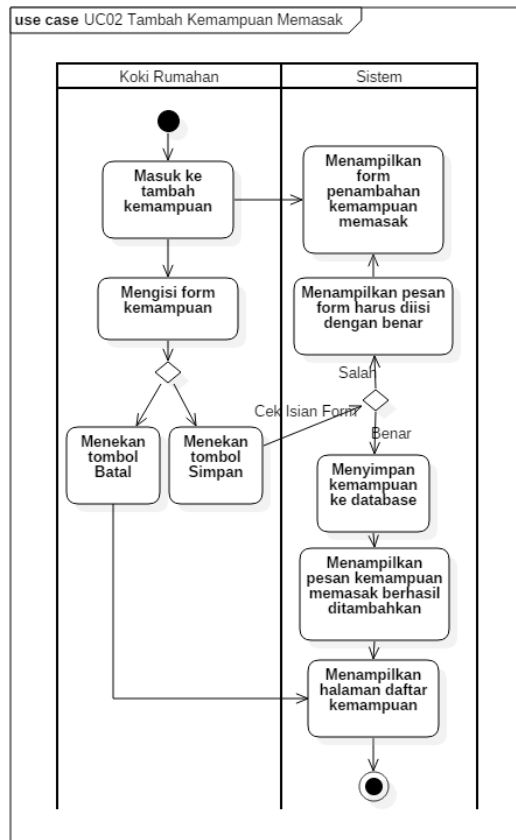
Pada Gambar 4.3 menjelaskan mengenai alur pendaftaran koki rumahan pada aplikasi. Koki rumahan menekan menu registrasi lalu akan muncul form pendaftaran. Kemudian Koki Rumahan mengisi datanya pada form pendaftaran, lalu menekan tombol register. Sistem akan melakukan validasi inputan, jika inputan telah sesuai maka sistem akan menampilkan pesan bahwa registrasi telah berhasil dan mengarahkan menuju ke halaman login. Jika inputan tidak sesuai maka, sistem akan menampilkan pesan bahwa form harus diisi dengan benar dan tetap menampilkan form pendaftaran.



Gambar 4.3 AD01 - Registrasi Diri

4.2.2.2 Tambah Kemampuan Memasak

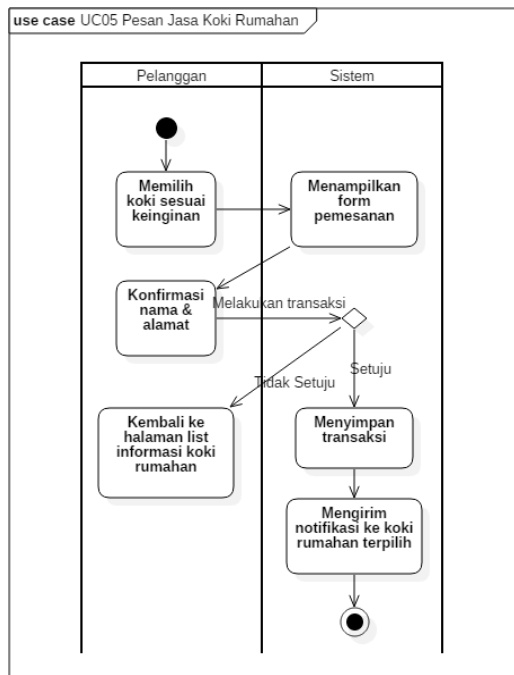
Pada Gambar 4.4 menjelaskan mengenai alur dalam menambahkan kemampuan memasak koki rumahan. Koki rumahan masuk ke dalam halaman untuk menambahkan kemampuan memasak lalu akan muncul form untuk menambahkan kemampuan memasak. Kemudian Koki Rumahan mengisi form kemampuan memasak. Kemudian Koki Rumahan mengisikan kemampuan memasaknya pada form tersebut.



Gambar 4.4 AD02 - Tambah Kemampuan Memasak

Terdapat dua tombol pada form tersebut yaitu tombol untuk menyimpan dan tombol untuk membatalkan. Ketika Koki Rumahan menekan tombol simpan, maka sistem akan melakukan validasi inputan, jika inputan telah sesuai maka sistem akan menampilkan pesan bahwa kemampuan memasak telah berhasil ditambahkan dan mengarahkan menuju ke halaman daftar kemampuan memasak. Jika inputan tidak sesuai maka, sistem akan menampilkan pesan bahwa form harus diisi dengan benar dan tetap menampilkan form penambahan kemampuan memasak. Sebaliknya, jika Koki Rumahan menekan tombol batal, maka sistem tidak akan melakukan validasi inputan dan langsung mengarahkan menuju halaman daftar kemampuan memasak.

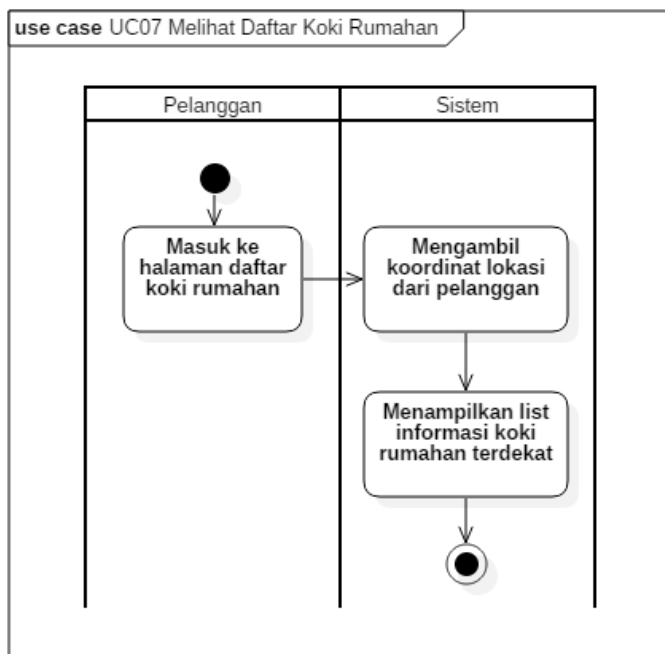
4.2.2.3 Pemesanan Jasa Koki Rumahan



Gambar 4.5 AD05 - Pemesanan Jasa Koki Rumahan

Pada Gambar 4.5 menjelaskan mengenai alur pemesanan jasa koki rumahan. Koki rumahan dapat dipilih sesuai dengan kriteria dari pelanggan. Setelah pelanggan memilih koki rumahan yang sesuai maka akan muncul form pemesanan. Kemudian pelanggan harus melakukan konfirmasi nama dan alamat yang akan digunakan sebagai tempat memasak. Jika pelanggan setuju untuk melakukan transaksi, maka sistem akan menyimpan transaksi tersebut dan sistem akan secara otomatis mengirimkan notifikasi kepada koki rumahan yang terpilih tersebut. Namun jika pelanggan tidak setuju untuk mengadakan transaksi, maka pelanggan akan diarahkan untuk menuju ke halaman daftar informasi koki rumahan.

4.2.2.4 Melihat Daftar Koki Rumahan

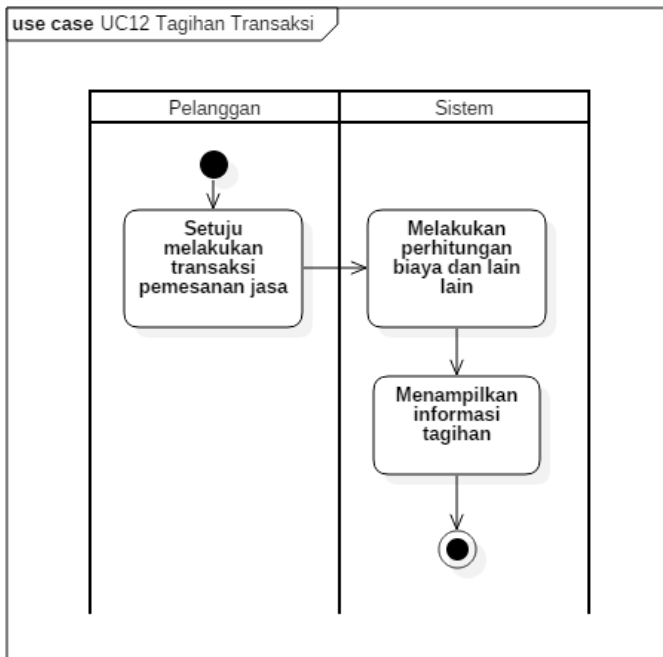


Gambar 4.6 AD07 - Melihat Daftar Koki Rumahan

Pada Gambar 4.6 menjelaskan mengenai alur untuk melihat daftar koki rumahan. Pelanggan mendapatkan rekomendasi daftar koki rumahan sesuai dengan lokasi pelanggan saat itu.

4.2.2.5 Tagihan Transaksi

Pada Gambar 4.7 menjelaskan mengenai alur untuk mendapatkan tagihan transaksi. Pelanggan mendapatkan tagihan transaksi pada saat setuju dalam melakukan transaksi.



Gambar 4.7 AD12 - Tagihan Transaksi

Setelah pelanggan setuju, maka sistem akan melakukan perhitungan biaya dan lain-lain. Kemudian sistem akan menampilkan informasi tagihan kepada pelanggan.

4.3 Analisis Kebutuhan Iterasi 2

Pengembangan aplikasi dengan menggunakan metode *prototyping*, membutuhkan beberapa kali perulangan. Pada saat

telah dilakukan implementasi pada iterasi pertama, pengguna aplikasi meminta beberapa fitur untuk ditambahkan ke dalam aplikasi. Fitur-fitur yang diminta tidak semuanya diterima namun dilakukan prioritas terlebih dahulu apakah sesuai dengan kebutuhan dari koki rumahan. Jika keinginan dari koki rumahan sesuai dengan kebutuhan, maka akan dimasukkan ke dalam fitur yang akan ditambahkan. Berikut ini beberapa perubahan-perubahan yang terjadi pada iterasi ke-2.

4.3.1 Kebutuhan Fungsional Sistem Iterasi 2

Perubahan kebutuhan fungsional sistem pada hasil iterasi ke-2 adalah seperti yang ditunjukkan pada Tabel 4.9 berikut ini:

Tabel 4.9 Perubahan Kebutuhan Fungsional Iterasi 2

ID	Status	ID User Need
KF03	Menambahkan sistem rating dan komentar ke dalam diskripsi informasi koki rumahan.	UN008, UN009
KF04	Ditambahkan dengan pengirim rincian tagihan ke email.	UN012

Dari hasil iterasi ke-2, pengguna aplikasi membutuhkan sebuah fungsi dan fitur baru. Fitur pertama yang ditambahkan adalah sistem *rating* dan komentar ke dalam informasi koki rumahan. Fitur ini digunakan untuk mengetahui kualitas dari koki rumahan. Kualitas ini dapat dijadikan sebagai salah satu bahan pertimbangan bagi pelanggan dalam memilih koki rumahan. Selain itu juga, koki mendapatkan *feedback* dan masukan dari pelanggan agar koki rumahan dapat memperbaiki kekurangannya. Terdapat juga fitur baru yang ditambahkan adalah fitur pengiriman rincian tagihan ke email. Fitur ini digunakan untuk memudahkan pelanggan maupun koki rumahan dalam mengakses rincian tagihan transaksi. Fitur ini dimasukkan ke dalam kebutuhan fungsional yang baru karena koki rumahan yang terbiasa dengan internet dan *gadget* serta menggunakan email dalam kesehariannya. Oleh karena itu, agar

koki rumahan tetap mendapatkan pemberitahuan, koki tidak harus sering untuk membuka web untuk memantau pesanan namun akan terdapat pemberitahuan secara otomatis melalui email.

4.3.2 Daftar *Use Case* Iterasi 2

Pada Tabel 4.10 berikut ini adalah tambahan dan perubahan *use case* pada iterasi ke-2. Perubahan ini didapatkan dari kebutuhan fungsional iterasi ke-2 yang kemudian diturunkan menjadi *use case* iterasi ke-2 yang baru.

Tabel 4.10 Daftar Perubahan *Use Case* Iterasi 2

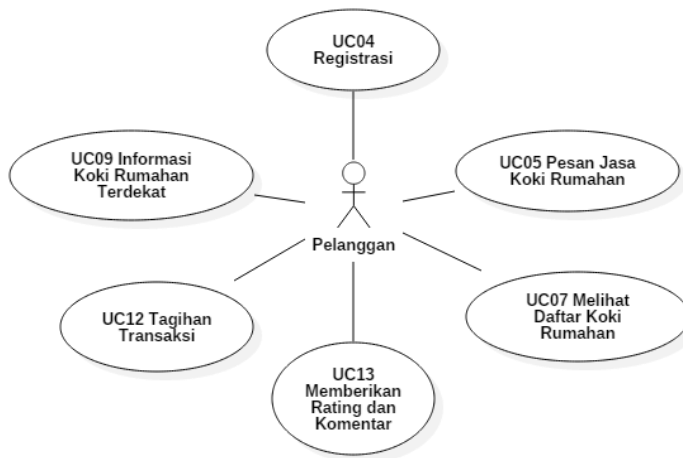
ID KF	ID Use Case	Use Case
KF03	UC13	Pelanggan dapat memberikan rating dan komentar kepada koki rumahan.
KF04	UC12	Pelanggan mendapatkan tagihan transaksi melalui email.

4.4 Perancangan Aplikasi Iterasi 2

Berikut ini adalah perancangan aplikasi pada iterasi kedua yang meliputi *Use Case* dan *Activity Diagram*.

4.4.1 Use Case Diagram Iterasi 2

Berikut ini adalah daftar *use case* diagram pada perubahan-perubahan pada iterasi kedua.



Gambar 4.8 *Use Case* Diagram Aktor Pelanggan Iterasi 2

Seperti yang terlihat pada Gambar 4.8 iterasi kedua pada aktor Pelanggan mendapatkan *use case* baru yaitu memberikan *rating* dan komentar. Pemberian *rating* dan komentar digunakan untuk memberikan *feedback* kepada koki rumahan mengenai masakannya, selain itu juga *rating* dan komentar dapat dijadikan tolak ukur bagi pelanggan dalam menentukan koki rumahan yang sesuai dengan kebutuhannya. Dan juga terdapat penambahan aktivitas pada *use case* tagihan transaksi yaitu mengirimkan tagihan transaksi ke email pelanggan. Pengiriman tagihan transaksi ini ditambahkan karena sesuai dengan kebutuhan calon pengguna aplikasi yang sering menggunakan email dalam kesehariannya.

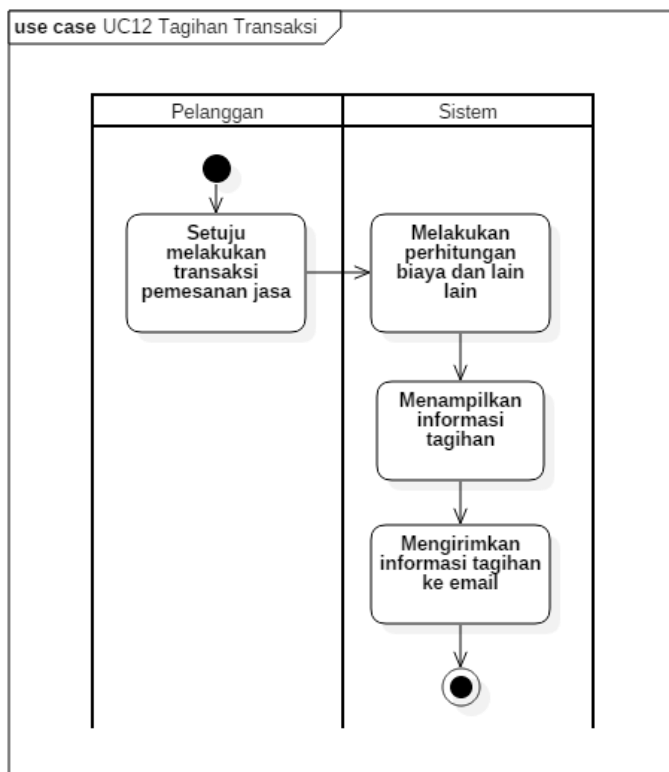
4.4.2 Diagram Aktivitas Iterasi 2

Berikut ini adalah daftar diagram aktivitas pada perubahan-perubahan pada iterasi kedua.

4.4.2.1 Tagihan Transaksi

Pada tagihan transaksi di iterasi kedua terjadi penambahan aktivitas pada sistem yaitu mengirimkan informasi tagihan ke email. Aktivitas ini ditambahkan karena pengguna aplikasi

sering menggunakan email dalam sehari-hari. Oleh karena itu fitur untuk mendapatkan informasi tagihan dapat ditambahkan karena sesuai dengan kebutuhan calon pengguna aplikasi. Sehingga calon pengguna aplikasi dapat dengan mudah mendapatkan pemberitahuan tagihan transaksi melalui email tanpa harus mengakses aplikasi setiap saat. Diagram aktivitas untuk tagihan transaksi dapat dilihat pada Gambar 4.9.

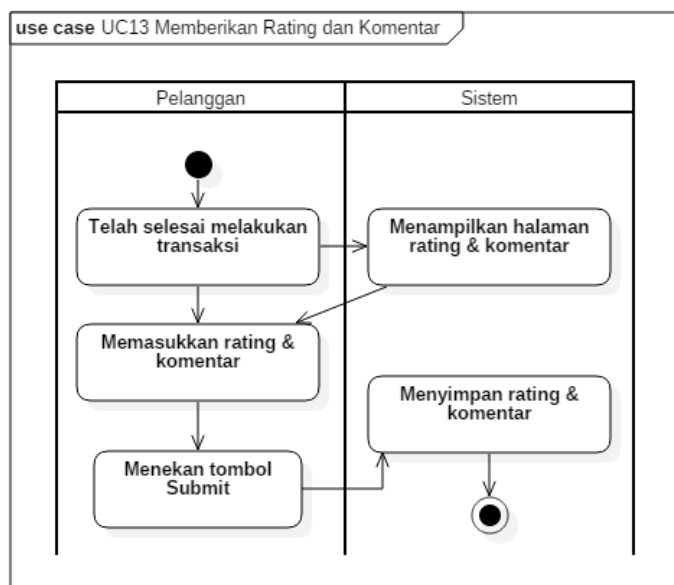


Gambar 4.9 AD12 - Tagihan Transaksi Iterasi 2

4.4.2.2 Memberikan Rating dan Komentar

Pada Gambar 4.10 menjelaskan mengenai alur dalam pemberian rating dan komentar kepada koki rumahan. Setelah transaksi dan proses pembayaran telah dilakukan, maka sistem

akan menampilkan halaman untuk memberikan rating dan komentar terhadap pelayanan jasa koki rumahan.



Gambar 4.10 AD13 - Memberikan Rating dan Komentar

Setelah selesai memasukkan *rating* dan komentar, lalu menekan tombol submit dan sistem akan menyimpan hasil inputan dari pelanggan tersebut. Hasil *rating* dan komentar tersebut akan ditampilkan pada halaman informasi koki rumahan untuk menjadi bahan evaluasi bagi koki tersebut.

4.5 Protokol Pengujian Aplikasi

Dalam melakukan *usability testing*, terdapat beberapa langkah yang harus dilakukan untuk mendapatkan hasil sesuai dengan kebutuhan yang telah didefinisikan. Pada penelitian ini protokol yang akan dijadikan sebagai acuan atau panduan dalam melakukan pengujian digambarkan ke dalam *Usability Test Plan Dashboard*. Berikut ini merupakan detail dari protokol pengujian yang digunakan untuk mendapatkan masukan dari pengguna yang ditunjukkan pada Tabel 4.11.

Tabel 4.11 Protokol Pengujian Aplikasi

<i>Domain</i>	<i>Description</i>
<i>Author</i>	Faisal Setia Putra
<i>Contact Detail</i>	faisalsetiap@gmail.com
<i>Product Under Test</i>	Website Aplikasi pencarian koki rumahan yang terdekat dan tersedia. Aplikasi ini bertujuan untuk menghubungkan antara koki rumahan dengan individu.
<i>Business Case</i>	Pengujian ini bertujuan untuk mendapatkan masukan untuk pengembangan aplikasi selanjutnya
<i>Test Objectives</i>	<div><div>1. Apakah pengguna memahami tujuan dibuatnya aplikasi?</div><div>2. Apakah fitur yang disediakan telah sesuai dengan kebutuhan pengguna?</div></div>

<i>Domain</i>	<i>Description</i>
	<ol style="list-style-type: none"> 3. Apakah pengguna merasa nyaman dalam menggunakan aplikasi? 4. Apakah dengan adanya layanan berdasarkan lokasi dapat menjadi salah satu faktor penentu dalam memilih koki rumahan?
<i>Participants</i>	2 orang yang pernah membantu orang lain memasak dan diberi upah. 2 orang yang paham teknologi dan 2 orang awam. Partisipan terdiri dari laki-laki maupun perempuan.
<i>Equipment</i>	<ol style="list-style-type: none"> 1. Laptop dan aplikasi yang terinstall 2. Form responden 3. <i>Access point</i> untuk mengakses internet
<i>Test Tasks</i>	<ul style="list-style-type: none"> ✓ Masuk sebagai pengguna ✓ Mendaftarkan diri sebagai koki rumahan ✓ Melakukan pemesanan koki rumahan ✓ Menambahkan daftar kemampuan memasak ✓ Mengubah daftar kemampuan memasak ✓ Mencari koki berdasarkan kemampuan memasak ✓ Mengubah lokasi tempat tinggal

<i>Domain</i>	<i>Description</i>
	<ul style="list-style-type: none"> ✓ Melihat riwayat pemesanan jasa ✓ Mencetak tagihan pemesanan jasa ✓ Memberikan rating & komentar kepada koki rumahan ✓ Keluar dari aplikasi
<i>Responsibilities</i>	Faisal Setia Putra (PM, Moderator, client contact)
<i>Location & Dates</i>	Senin tanggal 02 Januari 2017. Lokasi yang diambil berada di daerah Simo Gunung. Aplikasi dapat diakses di alamat http://localhost/ta
<i>Procedure</i>	<ol style="list-style-type: none"> 1. Perkenalan identitas diri & tujuan 2. Pre-test Review 3. Pelaksanaan Test Task 4. Pengisian kuisisioner 5. Post-test Interview 6. Memberikan ucapan terima kasih

BAB V

IMPLEMENTASI

Bab implementasi ini menjelaskan bagaimana tahap-tahap penelitian diimplementasikan. Pada bab ini juga menjelaskan mengenai cara melakukan penelitian secara teknis agar dapat dilakukan kembali dengan mudah. Hal-hal yang akan dibahas pada bab ini adalah lingkungan implementasi, konfigurasi sistem, *user interface*, serta kode program.

5.1 Lingkungan Implementasi

Aplikasi ini dikembangkan dengan menggunakan beberapa perangkat keras diantaranya adalah handphone, tablet dan laptop. Pada tahapan ini terdapat dua poin dalam implementasi lingkungan, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*).

5.1.1 Implementasi Hardware

Lingkungan implementasi pada hardware yang digunakan untuk mengelola aplikasi pencarian koki rumahan terdekat dan tersedia menggunakan sebuah server komputer. *Hardware* yang digunakan dalam komputer server tersebut adalah sebuah prosesor Intel i7-6700HQ 2.60 Ghz dan RAM sebesar 8 GB seperti terlihat pada Tabel 5.1.

Tabel 5.1 Spesifikasi *Hardware* Server

Perangkat Keras	Spesifikasi
NoteBook	Processor: Intel Core i7 (2,60 Ghz)
	Memori: 8 GB
	Sistem Operasi: Windows 10

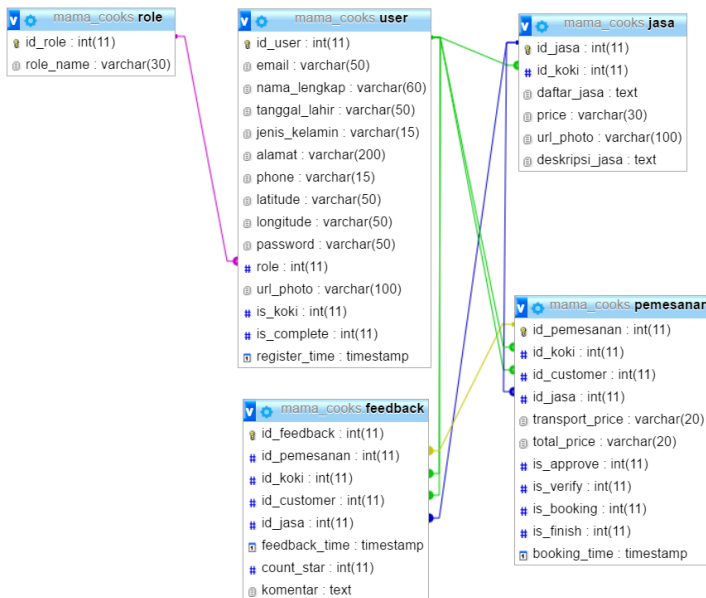
5.1.2 Implementasi Software

Spesifikasi *software* harus mampu menjalankan web browser. Sistem operasi yang minimal yang disarankan menggunakan windows 7 karena telah tersedia fasilitas web browser seperti

Google Chrome, Mozilla dan Internet Explorer yang mendukung javascript dan *geolocation*. *Geolocation* digunakan untuk mendapatkan lokasi posisi dari pengguna aplikasi.

5.2 Pembuatan Database

Setelah dilakukan analisis kebutuhan sistem serta hasil perancangan aplikasi, selanjutnya adalah membuat desain basis data pada sistem. Database dibuat berdasarkan desain model data fisik. Pembuatan model database pada aplikasi pencarian koki rumahan ini menggunakan database MySQL. Desain Database dapat dilihat pada Gambar 5.1



Gambar 5.1 Desain Database Aplikasi Pencarian Koki

5.3 Implementasi Fungsi

Pengkodean dan implementasi fungsi dilakukan setelah pembuatan database. Seluruh fitur yang dikembangkan atau dibuat telah melalui proses iterasi sebanyak dua kali. Pada pengkodean aplikasi dapat dilihat pada lampiran D, peneliti menggunakan bahasa pemrograman PHP dan dengan database

MySQL. Pada pembuatan aplikasi ini terdapat berbagai fitur yang diimplementasikan, berikut ini merupakan salah satu contoh implementasi fungsi:

5.3.1 Implementasi Login

Pada saat pertama kali membuka aplikasi, maka pengguna akan dihadapkan dengan halaman awal. Pengguna dapat mengakses aplikasi pencarian koki rumahan, jika pengguna telah melakukan pendaftaran pada aplikasi terlebih dahulu. Setelah pengguna melakukan pendaftaran, maka pengguna dapat masuk ke dalam aplikasi dengan membuka halaman *Login*.

```

1  $email = $_POST['email'];
2  $password =
    sha1(mysql_real_escape_string($conn,($_P
    OST['password'])));

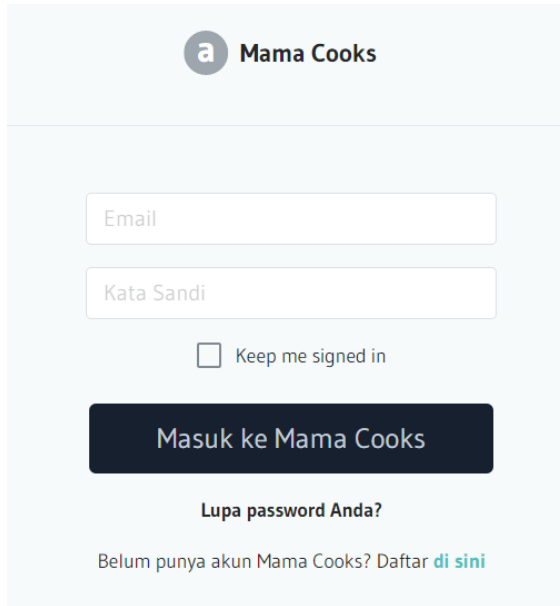
3  $sql = "SELECT * FROM `user` WHERE
    `email` = '$email' AND `password` =
    '$password'";
4  $result = $conn ->query($sql);

```

Gambar 5.2 Code Login Aplikasi

Terlihat pada **Error! Reference source not found.** pada baris ke 1-2 dilakukan pengambilan nilai dari inputan email dan password dimana password yang diketikkan di enkripsi menjadi format SHA1 dan dengan penambahan fungsi untuk mencegah terjadinya *SQL Injection*. Kemudian pada baris ke 3-4 dilakukan pengecekan data pada database, jika data terdapat pada database dan sesuai maka halaman akan diarahkan sesuai dengan peran dari pengguna. Jika pengguna memiliki peran sebagai Admin atau dengan nilai *role* sama dengan 0, maka pengguna akan diarahkan ke halaman admin. Jika nilai *role* sama dengan 1 atau 2, maka pengguna akan diarahkan menuju halaman *home*. Setelah itu pengguna akan mendapatkan *session* untuk setiap user yang datanya terdapat pada database. Jika data

pengguna tidak terdapat pada database aplikasi maka akan muncul pesan kesalahan. Setelah pesan kesalahan muncul, maka halaman akan dikembalikan menuju halaman sebelumnya.



Gambar 5.3 Tampilan Halaman *Login*

Pada tampilan Login juga disediakan tombol khusus untuk mengarahkan ke halaman pendaftaran diri pada aplikasi seperti terlihat pada Gambar 5.3. Tombol pendaftaran tersebut digunakan untuk mempermudah pengguna dalam melakukan pendaftaran akun tanpa harus mencari halaman pendaftaran.

5.3.2 Implementasi Pendaftaran Pelanggan

Sebelum menggunakan aplikasi, pengguna harus mendaftarkan identitas diri terlebih dahulu. Data-data yang harus diisi diantaranya adalah nama lengkap, jenis kelamin, email, dan password yang digunakan untuk menggunakan aplikasi. Seperti yang terlihat pada Gambar 5.4, isian dari nama lengkap, email,


```

1  $fullname = $_POST['fullname'];
2  $email = $_POST['email'];
3  $gender = $_POST['jenis_kelamin'];
4  $password =
    sha1(mysql_real_escape_string($conn,($_P
    OST['password'])));

5  $sql = "SELECT email FROM `user` WHERE
    `email`='$email'";
6  $result = $conn ->query($sql);

7  if(isset($_POST['submit'])){
8      if(mysql_num_rows($result)>0){
9          echo "<script>alert('Maaf Email telah
            terdaftar');
            window.history.back();</script>";
10 } else {
11     $sql = "INSERT INTO
        `user`(`nama_lengkap`, `email`,
        `jenis_kelamin`, `password`, `role`)
        VALUES ('$fullname', '$email', '$gender',
        '$password', '1')";
12     $result = $conn ->query($sql);

```

Gambar 5.4 Code Pendaftaran Pelanggan

jenis kelamin dan password dimasukkan ke dalam database. Sebelum pengguna mendaftarkan diri, hal yang diperiksa pertama kali adalah alamat email. Pada baris ke 8-9 merupakan fungsi untuk memeriksa apakah terdapat email yang sama di database dengan email yang didaftarkan oleh pengguna. Jika pengguna menggunakan alamat email yang sama dengan alamat email yang telah terdaftar pada aplikasi, maka aplikasi akan memunculkan pesan peringatan email telah terdaftar. Jika email belum terdaftar di database, maka sistem akan menjalankan

query pada baris ke-11 hingga baris ke-12 yang merupakan *query* untuk memasukkan data ke dalam database. Setelah itu, aplikasi akan menampilkan pesan bahwa pengguna telah berhasil terdaftar.

di sini'."/>

Gambar 5.5 Tampilan Halaman Pendaftaran Pelanggan

Seperti yang terlihat pada Gambar 5.5, dengan mendaftarkan diri maka pengguna telah setuju terhadap Syarat dan Ketentuan serta Kebijakan Privasi dari aplikasi Mama Cooks. Selain itu, sistem juga telah menyediakan validasi isian yang memastikan semua isian harus terisi sebelum melakukan pendaftaran.

5.3.3 Implementasi Pendaftaran Koki Rumahan

Seperti yang terlihat pada Gambar 5.6, koki rumahan dapat mendaftarkan diri dengan memasukkan kemampuan memasak, deskripsi jasa, dan tarif yang dikenakan untuk pemesanan

```

1  $skill = $_POST['skill'];
2  $deskripsi = $_POST['deskripsi'];
3  $price = $_POST['price'];

4  if(isset($_POST['submit'])){
5  if(isset($_POST['check'])){
6  $sql = "INSERT INTO `jasa`(`id_koki`,
`daftar_jasa`, `price`, `deskripsi_jasa`)
VALUES ('$id_user', '$skill', '$price',
'$deskripsi')";
7  $result = $conn ->query($sql);
8  if($result) {
9  $sql_update = "UPDATE `user` SET
`is_koki`='1', `role`='2' WHERE
`id_user`='$id_user'";
10 $result_update = $conn-> query
($sql_update);
11 echo " <script>alert('Selamat, Anda
berhasil terdaftar menjadi koki
rumahan');
12 window.location.href = 'home';
13 </script>";
14 }

```

Gambar 5.6 *Code* Pendaftaran Koki Rumahan

seperti yang terdapat pada baris ke-1 hingga baris ke-3. Pada baris ke-5 terdapat validasi inputan check, jika koki rumahan tidak setuju terhadap kebijakan layanan, maka data tidak akan tersimpan. Sebaliknya, jika koki rumahan setuju maka sistem akan menyimpan data-data tersebut ke dalam database aplikasi dan secara otomatis akan menampilkan informasi tersebut pada halaman daftar koki rumahan. Setelah itu, sistem akan menampilkan pesan bahwa koki rumahan berhasil terdaftar. Untuk pendaftaran koki rumahan dilakukan pada saat pengguna telah mendaftarkan diri menjadi pelanggan. Setelah pengguna

mendaftarkan diri pada aplikasi, maka akan muncul tombol untuk melakukan registrasi menjadi koki rumahan. Setelah pengguna mendaftarkan diri menjadi koki rumahan, maka akan muncul menu baru pada halaman pengaturan yang digunakan untuk mengganti kemampuan memasak maupun harga yang ditetapkan untuk sekali pemesanan jasa.

Pendaftaran Koki Rumahan

Isi dengan benar.

Keahlian Memasak Makanan

Pisahkan dengan tanda koma. Contoh: Rawon, Pecel, Sate

Deskripsi Jasa

Terima jasa catering

Harga per Jasa

150000

Gambar Pendukung

Masukkan gambar pendukung. Contoh: Gambar Hasil Masakan

Letakkan file disini atau klik untuk *upload*.

(Masukkan semua gambar pendukung disini.)

Gambar 5.7 Tampilan Halaman Pendaftaran Koki Rumahan

Pada Gambar 5.7 merupakan tampilan antar muka untuk melakukan pendaftaran menjadi koki rumahan. Terdapat beberapa isian-isian diantaranya adalah keahlian memasak makanan. Pada isian tersebut, koki rumahan dapat memasukkan semua jenis makanan yang dapat diolah untuk menjadi makanan. Selain itu, terdapat juga isian deskripsi jasa, isian

tersebut dapat digunakan sebagai data pendukung semisal keahlian memasak yang telah tersertifikasi. Pada isian harga per jasa diisi dengan harga pemesanan jasa untuk memasak.

5.3.4 Implementasi Kemampuan Memasak

Pada Gambar 5.8 merupakan *source code* untuk mengubah kemampuan memasak. Pada baris ke-1 hingga baris ke-2 terlihat *query* untuk melakukan perubahan kemampuan memasak, deksripsi dan juga harga untuk setiap pemesanan yang dapat diubah sesuai dengan kebutuhan. Pada baris ke-5 merupakan validasi terhadap ketersediaan jasa koki. Jika koki rumahan tidak mengaktifkan status ketersediaan untuk pemesanan, maka sistem tidak akan menampilkan informasi

```

1  $sql = "UPDATE `jasa` SET
    `daftar_jasa`='$skill', `price`='$price',
    `deskripsi_jasa`='$deskripsi' WHERE
    id_koki='$id_user';";
2  $result = $conn ->query($sql);

3  if(isset($_POST['submit'])) {
4  if($result) {
5  if (isset($_POST['available'])) {
6  $sql = "UPDATE `user` SET `is_koki`='1'
    WHERE id_user='$id_user'";
7  $result = $conn ->query($sql);
8  } else {
9  $sql = "UPDATE `user` SET `is_koki`='0'
    WHERE id_user='$id_user'";
10 $result = $conn ->query($sql);
11 }}

```

Gambar 5.8 Code Mengubah Kemampuan Memasak Koki

koki rumahan tersebut pada daftar koki rumahan yang tersedia untuk melayani pelanggan. Jika koki rumahan tersebut mengaktifkan status ketersediaan, maka sistem akan secara

otomatis menampilkan informasi koki rumahan tersebut pada daftar koki rumahan yang dapat dipesan oleh pelanggan. Dengan mendaftar sebagai koki rumahan, maka pengguna aplikasi dapat menambahkan, mengurangi maupun mengubah kemampuan dalam mengolah makanan. Selain itu koki rumahan juga dapat menentukan ketersediaan mereka. Jika mereka tidak memiliki waktu untuk melayani pelanggan, maka mereka dapat mematikan fitur untuk melayani pelanggan. Jika mereka ingin menerima pesanan, maka fitur untuk melayani pelanggan dapat diaktifkan kembali. Fitur ini ditambahkan dikarenakan tidak semua koki rumahan dapat melayani pelanggan pada saat koki rumahan berhalangan untuk melayani seperti saat koki sakit maupun ada urusan lain yang mendadak.

Keahlian

Keahlian Memasak Makanan

Rawon, Chicken Cordon Bleu, Pecel, Sate, Nasi Goreng

Deskripsi Jasa

Terima jasa catering dan acara besar

Harga per Jasa

50000

Gambar Pendukung

Masukkan gambar pendukung. Contoh: Gambar Hasil Masakan

Letakkan file disini atau klik untuk *upload*.

(Masukkan semua gambar pendukung disini.)

Gambar 5.9 Tampilan Halaman Mengubah Kemampuan Memasak

Koki rumahan dapat mengubah data-data yang berkaitan dengan jasanya seperti keahlian memasak makanan. Keahlian dalam memasak makanan dapat diisi dengan berbagai macam jenis masakan namun setiap masakan dipisahkan dengan tanda koma. Selain itu, koki rumahan juga dapat mengubah deskripsi pendukung jasa koki tersebut seperti yang terlihat pada Gambar 5.9. Koki juga dapat menentukan biaya tarif yang dikenakan dalam sekali pemesanan jasa.

5.3.5 Implementasi Pemesanan Koki

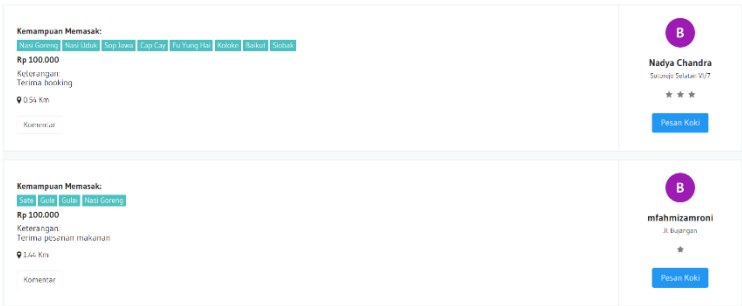
```

1  $id_jasa = $_GET['jasa'];
2  $id_koki = $_GET['user'];
3  $distance = $_GET['d'];
4  $sql_check = "SELECT * FROM `pemesanan`
   WHERE id_jasa='$id_jasa' AND
   is_verify='0' OR is_verify='';
5  $result_check = $conn->query($sql_check);
6  if(mysqli_num_rows($result_check)){
7  $sql_jasa = "SELECT * FROM `jasa` AS j,
   `user` AS u WHERE j.id_jasa='$id_jasa'
   AND u.id_user='$id_koki'";
8  $result_jasa = $conn->query($sql_jasa);
9  $row_jasa = $result_jasa->fetch_assoc();
10 $harga = $row_jasa['price'];
11 $id_koki = $row_jasa['id_koki'];
12 } else { $sql_jasa = "SELECT * FROM
   `jasa` AS j, `user` AS u WHERE
   j.id_jasa='$id_jasa' AND
   u.id_user='$id_koki'";
13 $result_jasa = $conn->query($sql_jasa);
14 $row_jasa = $result_jasa->fetch_assoc();
15 $harga = $row_jasa['price'];
16 $id_koki = $row_jasa['id_koki'];
17 $sql = "INSERT INTO `pemesanan`
   (`id_koki`, `id_customer`, `id_jasa`,
   `transport_price`, `total_price`,
   `is_verify`, `is_approve`, `is_booking`,
   `is_finish`) VALUES ('$id_koki',
   '$id_user', '$id_jasa', '$transport',
   ($harga+$transport), '0', '0', '0',
   '0')"; $result = $conn->query($sql); }

```

Gambar 5.10 *Code Pemesanan Koki*

Pada halaman pemesanan koki seperti yang terlihat pada Gambar 5.10 , sistem mengambil nilai id jasa, id user dan jarak yang ditempuh dari lokasi koki rumahan menuju lokasi pelanggan seperti yang terdapat pada baris ke-4 hingga baris ke-6. Dari harga pemesanan jasa ditambah dengan jarak lokasi maka dapat ditentukan biaya yang harus dikeluarkan oleh pemesan kepada koki rumahan. Biaya tersebut terdapat pada halaman tagihan pelanggan yang secara otomatis akan dikirim melalui email oleh sistem. Ketika pelanggan setuju untuk melakukan transaksi, maka pemberitahuan pemesanan akan terkirim kepada koki rumahan. Kemudian, koki rumahan dapat memilih untuk menerima pesanan tersebut atau menolaknya. Jika koki rumahan tersebut menolak ataupun menyetujui pemesanan, maka pelanggan akan langsung mendapatkan pemberitahuan tersebut. Setiap transaksi pemesanan, koki rumahan hanya dapat melayani satu pelanggan saja. Sebelum pelanggan tersebut melakukan verifikasi bahwa pekerjaan koki rumahan telah selesai, maka koki rumahan tidak dapat muncul pada daftar koki yang dapat dipesan. Setelah pelanggan melakukan verifikasi, maka koki rumahan dapat menerima pesanan kembali.



Gambar 5.11 Tampilan Halaman Pemesanan Koki Rumahan

Selain itu, koki rumahan juga dapat menolak pesanan dari pelanggan. Jika koki rumahan menolak pesanan dari pelanggan, maka koki rumahan tersebut akan ditampilkan kembali pada

halaman pesanan. Tampilan halaman pemesanan koki rumahan dapat dilihat seperti pada Gambar 5.11.

5.3.6 Implementasi Informasi Tagihan

```

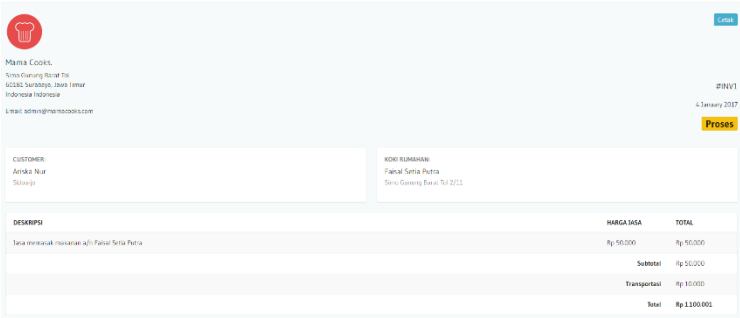
1  $id = $_GET['inv'];
2  $sql = "SELECT * FROM `pemesanan` WHERE
    id_pemesanan='$id'";
3  $result = $conn->query($sql);
4  $row = $result->fetch_assoc();
5  if($row['is_booking']=='0' &&
    $row['is_verify']=='1' &&
    $row['is_approve']=='0'){ ?>
6  <p><span class="label label-lg
    danger">Ditolak</span></p><br>
7  <?php } else if ($row['is_approve']=='1'
    && $row['is_booking']=='0' &&
    $row['is_finish']=='1'){?>
8  <p><span class="label label-lg
    success">Lunas</span></p><br>
9  <?php } else {?>
10 <p><span class="label label-lg
    warning">Proses</span></p><br>
11 <?php } ?>

```

Gambar 5.12 Code Informasi Tagihan

Pada Gambar 5.12 terdapat *query* pada baris ke-2 untuk menampilkan jumlah tagihan. Tagihan tersebut didapatkan berdasarkan dari id pemesanan yang telah dilakukan. Selain itu juga terdapat *code* pada baris ke-5 hingga baris ke-11 yang digunakan untuk memberikan label pada tagihan. Label tersebut diantaranya adalah Ditolak, Proses, dan Lunas. Parameter untuk menentukan tagihan tersebut masuk dalam kategori Ditolak, Proses dan Lunas adalah berdasarkan dari pelanggan dan juga dari koki rumahan. Jika pelanggan melakukan verifikasi bahwa koki telah melakukan pekerjaannya maka tagihan akan secara

otomatis diberi label Lunas. Namun, jika koki rumahan tersebut menolak pesanan, maka tagihan akan secara otomatis diberi label Ditolak.



Gambar 5.13 Tampilan Halaman Informasi Tagihan

Tampilan antar muka dari halaman informasi tagihan dapat dilihat pada Gambar 5.13. Pada halaman informasi tagihan terdapat informasi mengenai pelanggan yang melakukan pemesanan dan juga terdapat informasi mengenai koki rumahan yang di pesan. Selain itu, terdapat juga deskripsi pemesanan jasa dan total harga pemesanan jasa. Laporan tagihan dapat disimpan dalam bentuk PDF, selain itu juga laporan tagihan dapat langsung dicetak.

5.3.7 Implementasi Koordinat Lokasi

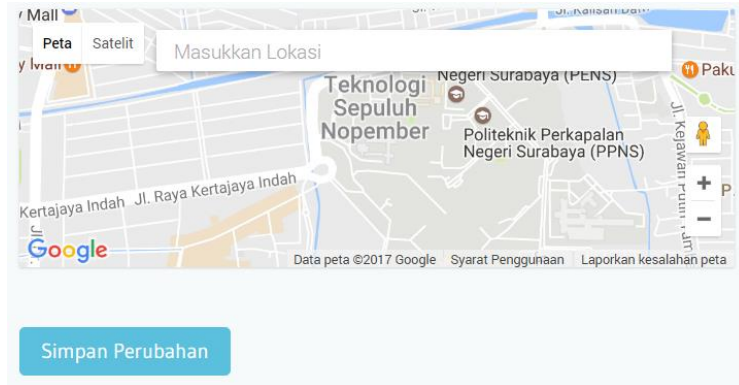
Seperti yang terlihat pada Gambar 5.14, koordinat lokasi diambil dari lokasi pengguna mengakses aplikasi. Aplikasi akan secara otomatis menampilkan map yang berada di sekitar lokasi pengakses aplikasi. Isian latitude dan longitude akan secara otomatis terisi oleh sistem pada saat pengguna melakukan pencarian lokasi menggunakan kolom pencarian lokasi. Kode untuk mengisi latitude dan longitude secara otomatis terdapat pada kode baris ke-1 hingga baris ke-4.

```
1  google.maps.event.addListener(marker,  
   "mouseup", function(event) {  
2  $('#input-  
   latitude').val(this.position.lat());  
3  $('#input-  
   longitude').val(this.position.lng());  
4  });  
5  function showError(error) {  
6  switch(error.code) {  
7  case error.PERMISSION_DENIED:  
8  alert("User denied the request for  
   Geolocation.");  
9  break;  
10 case error.POSITION_UNAVAILABLE:  
11 alert("Location information is  
   unavailable.");  
12 break;  
13 case error.TIMEOUT:  
14 alert("The request to get user location  
   timed out.");  
15 break;  
16 case error.UNKNOWN_ERROR:  
17 alert("An unknown error occurred.");  
18 break;  
19 }
```

Gambar 5.14 *Code Koordinat Lokasi*

Kolom pencarian lokasi akan menampilkan daftar nama-nama lokasi berdasarkan isian pengguna. Jika browser menolak permintaan aplikasi untuk mengakses lokasi, maka sistem akan menampilkan informasi mengenai penolakan permintaan tersebut. Untuk menangani hal tersebut maka ditambahkan sebuah fungsi `showError` yang dapat dilihat pada baris ke-5

hingga baris ke-19. Selain penolakan permintaan untuk mengakses lokasi, informasi akan muncul jika lokasi dari pengakses aplikasi tidak ditemukan.



Gambar 5.15 Tampilan Halaman Koordinat Lokasi

Tampilan peta diambil dari Google Maps API seperti yang terlihat pada Gambar 5.15. Selain itu, pengguna aplikasi dapat menentukan sendiri lokasi tempat tinggal dengan menggeser titik ke lokasi tempat yang diinginkan.

5.3.8 Implementasi Pencarian Koki Terdekat

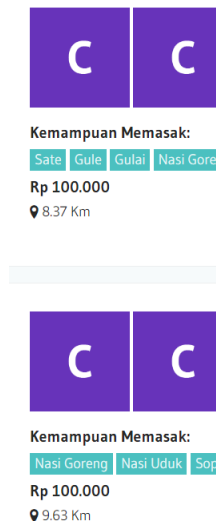
```

1 function showLocation(position) {
2   var latitude = position.coords.latitude;
3   var longitude = position.coords.longitude;
4   function getLocation(){
5     if(navigator.geolocation){var options =
      {timeout:60000};
6     navigator.geolocation.getCurrentPosition(showLocation, errorHandler, options);
7   } else { alert("Sorry, browser does not support geolocation!"); }

```

Gambar 5.16 Code Pencarian Koki Terdekat

Tampilan pada Gambar 5.16 merupakan *source code* dengan menggunakan javascript *Google Maps API*. Fungsi `showLocation` pada baris ke-1 hingga ke-3 merupakan fungsi untuk menampilkan lokasi pengguna saat ini. Dan fungsi `getLocation` pada baris ke-4 digunakan untuk menjalankan fungsi `showLocation` dengan penambahan kode untuk menangani masalah apabila browser tidak mendukung *geolocation*. Setelah posisi pengguna diambil, kemudian akan ditampilkan rekomendasi berdasarkan jarak antara koki rumahan dengan pengguna aplikasi.

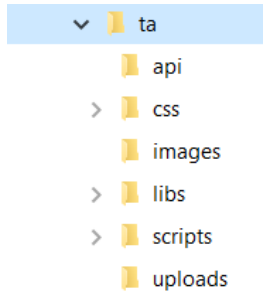


Gambar 5.17 Tampilan Rekomendasi Koki Berdasarkan Jarak

Penghitungan jarak koki dengan pengguna aplikasi diambil dari database *latitude* dan *longitude* koki rumahan. Setelah data tersebut diambil, maka sistem akan menghitung jarak antara *latitude* dan *longitude* dari pengguna aplikasi dengan koki rumahan. Setelah itu sistem akan melakukan pengurutan berdasarkan jarak seperti yang terlihat pada Gambar 5.17. Satuan yang digunakan sebagai jarak adalah satuan KM.

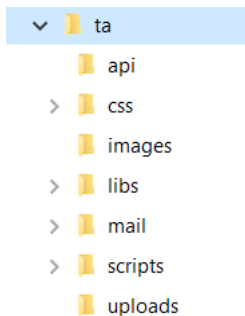
5.4 Struktur Direktori

Pada pengembangan aplikasi mengalami iterasi sebanyak 2 kali. Struktur direktori dari pengembangan aplikasi masih menggunakan struktur prosedural karena tidak menggunakan framework. Gambar 5.18 menunjukkan struktur direktori dari aplikasi pada iterasi ke-1.



Gambar 5.18 Struktur Direktori Aplikasi Iterasi Ke-1

Pada iterasi ke-2, terdapat penambahan library baru berupa direktori *mail*, direktori mail berisikan *library* agar aplikasi dapat mengirimkan pesan email kepada pelanggan maupun koki rumahan seperti yang terlihat pada Gambar 5.19. Email yang dikirimkan oleh aplikasi merupakan email yang berisikan tagihan pemesanan pada saat pelanggan setuju untuk melakukan pemesanan.



Gambar 5.19 Struktur Direktori Aplikasi Iterasi Ke-2

BAB VI

HASIL PENGUJIAN DAN PEMBAHASAN

Pengujian *prototype* dibuat berdasarkan fitur yang ada dalam aplikasi. Pada setiap tahapan proses, dilakukan perancangan pengujian dalam bentuk perencanaan dan prosedur pengujian untuk kebutuhan fungsional dan kebutuhan non-fungsional pada aplikasi. Hasil yang didapatkan telah melalui iterasi sebanyak dua kali menggunakan metode *Usability Testing*. Pada uji coba yang dilakukan, calon pengguna diminta untuk melakukan evaluasi terhadap seluruh fitur yang ada, memberikan kelebihan maupun kekurangan dari aplikasi. Selain itu, calon pengguna juga diminta untuk memberikan masukan untuk menentukan fitur-fitur apa saja yang perlu ditambahkan ataupun dikurangi. Pada iterasi yang dilakukan, seluruh komentar dan *user story* yang telah sesuai dengan kebutuhan diimplementasikan ke dalam *prototype* dengan menggunakan PHP. Detail dari perancangan pengujian ini dilampirkan pada dokumen pengujian.

6.1 *Usability Testing*

Pada pengembangan aplikasi, untuk mendapatkan banyak masukan, aplikasi diuji dengan membuat *Minimum Viable Product* (MVP) dari seluruh *user story* yang ada. *Usability testing* dilakukan dengan dua iterasi uji coba. Seluruh hasil dari *usability testing* yang dilakukan didokumentasikan pada dokumen protokol dan hasil uji coba.

6.1.1 *Usability Testing I*

Pada tahapan *usability testing* yang pertama, dilakukan pada tanggal 30 Desember 2016. *Usability testing* pada tahapan ini dilakukan dengan membuat aplikasi *Minimum Viable Product* (MVP) sesuai dengan kebutuhan-kebutuhan utama yang didapatkan dari *user story*. Setelah itu aplikasi diujikan langsung kepada koki rumahan untuk mengetahui kesesuaian dengan kebutuhan dari koki rumahan.

6.1.2 Usability Testing II

Pada tahapan ini, pengujian aplikasi kembali dilakukan untuk mendapatkan kembali masukan bagi pengembangan aplikasi. Pada *usability testing* yang kedua, penulis menggunakan aplikasi yang sebelumnya telah diperbaiki sesuai dengan masukan pada *usability testing* pertama. Pada pengujian ini digunakan *usability testing plan dashboard* yang terdapat pada Lampiran A *Usability Test Plan Dashboard*. Pada akhir setiap sesi *usability testing* yang dilakukan, responden diminta untuk mengisi kuisioner *System Usability Scale* (SUS) yang digunakan untuk mengukur kemudahan dari penggunaan aplikasi. Skenario *Test Case* dapat dilihat pada Lampiran G.

Tabel 6.1 Daftar *Test Case*

ID Test Case	Test Case
TC01	Masuk sebagai pengguna
TC02	Mendaftarkan diri sebagai koki rumahan.
TC03	Melakukan pemesanan koki rumahan
TC04	Menambahkan daftar kemampuan memasak
TC05	Mengubah daftar kemampuan memasak
TC06	Mencari koki berdasarkan kemampuan memasak
TC07	Mengubah lokasi tempat tinggal
TC08	Melihat riwayat pemesanan
TC09	Mencetak tagihan pemesanan jasa
TC10	Memberikan rating dan komentar kepada koki rumahan
TC11	Keluar dari aplikasi

Hasil dari kuisioner *System Usability Scale* menunjukkan aplikasi yang dikembangkan memiliki kemudahan dalam penggunaan sebesar 87,92% seperti yang terlihat pada Lampiran C. Pada kuisioner tersebut menunjukkan bahwa banyak pengguna yang setuju dengan pernyataan yang diajukan kepada mereka. Seperti yang terlihat pada Tabel 6.2.

Tabel 6.2 Hasil Justifikasi *System Usability Scale*

Pernyataan	Justifikasi
Pengguna berpikir bahwa saya ingin menggunakan sistem ini sesering mungkin	Setuju
Pengguna berpikir sistem ini sangat rumit dan tidak perlu	Tidak Setuju
Pengguna berpikir sistem ini mudah digunakan	Setuju
Pengguna berpikir saya akan membutuhkan bantuan seseorang untuk dapat menggunakan sistem ini	Tidak Setuju
Pengguna menemukan seluruh fungsi yang telah terintegrasi dengan sangat baik	Setuju
Pengguna berpikir terlalu banyak inkonsistensi dari sistem ini	Tidak Setuju
Pengguna berpikir banyak orang yang akan dapat belajar menggunakan sistem ini dengan cepat	Setuju
Pengguna berpikir bahwa sistem ini sangat sulit untuk digunakan	Tidak Setuju
Pengguna merasa sangat percaya diri dalam menggunakan sistem ini	Setuju
Pengguna perlu belajar banyak hal sebelum dapat menggunakan sistem ini	Tidak Setuju

Adapun hasil dari kuisioner diatas dapat dilihat pada Lampiran E untuk Lembar Persetujuan dan Lampiran F untuk kuisioner mengenai *System Usability Scale* (SUS).

Halaman ini sengaja dikosongkan

BAB VII

KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan mengenai kesimpulan dan saran dari tugas akhir sesuai dengan yang terdapat pada bab sebelumnya.

7.1 Kesimpulan

Tugas akhir ini adalah pembuatan aplikasi pencarian koki rumahan yang terdekat dan tersedia berbasis web. Dari hasil uji coba yang dilakukan pada penelitian ini, penulis mendapatkan kesimpulan sebagai berikut:

1. Aplikasi pencarian koki rumahan telah dibuat berdasarkan proses bisnis sehari-hari dan kebutuhan yang didapatkan dari wawancara yaitu:
 - a. Sistem menyediakan pendaftaran bagi koki rumahan dan pelanggan.
 - b. Sistem menyediakan pemesanan jasa memasak makanan koki rumahan.
 - c. Sistem menyediakan informasi koki rumahan berdasarkan lokasi terdekat.
 - d. Sistem harus menyediakan rincian tagihan transaksi pemesanan jasa koki rumahan.
2. Metode pengembangan yang digunakan untuk mengembangkan layanan berbasis lokasi untuk menemukan koki rumahan yang terdekat dan tersedia yaitu dengan metode *Prototyping* dengan 3 tahapan sebagai berikut:
 - a. Pengumpulan kebutuhan dari sistem dengan cara mendengar keluhan dari pelanggan.
 - b. Merancang dan membuat prototype dari sistem. Prototype yang dibuat disesuaikan dengan kebutuhan sistem yang telah didefinisikan sebelumnya dari keluhan pelanggan maupun pengguna.

- c. Melakukan uji coba oleh pelanggan maupun pengguna. Kemudian dilakukan evaluasi kekurangan-kekurangan maupun masukan dari kebutuhan pelanggan. Lalu dilakukan pengembangan kembali dengan melalui tahapan dari awal yaitu mendengarkan keluhan dari pelanggan untuk memperbaiki *prototype* yang telah ada
3. Aplikasi pencarian koki rumahan telah melewati skenario uji coba untuk semua kebutuhan fungsional dengan hasil yang ditampilkan pada sistem telah sesuai dengan kebutuhan.

7.2 Saran

Berdasarkan penelitian pada tugas akhir ini dapat diberikan saran ataupun masukan untuk pengembangan selanjutnya, antara lain:

1. Untuk penelitian selanjutnya dapat ditambahkan dengan pengiriman notifikasi dalam bentuk SMS kepada koki rumahan maupun pelanggan. Notifikasi dalam bentuk SMS ini digunakan karena aplikasi *mobile* sesuai jika digunakan.
2. Selain penambahan notifikasi, aplikasi dapat dikembangkan dengan penambahan fitur penjadwalan pemesanan koki rumahan. Penjadwalan ini digunakan untuk menjadwalkan koki pada saat terdapat pemesanan. Dan juga pelanggan dapat memesan koki jauh-jauh hari.
3. Aplikasi dapat dikembangkan lagi dengan berbasis android ataupun iOS agar pengguna lebih nyaman dan mudah dalam menggunakan aplikasi.

DAFTAR PUSTAKA

- [1] K. RI, Buku Pedoman Penyelidikan dan Penanggulangan Kejadian Luar Biasa Penyakit Menular dan Keracunan Pangan (Pedoman Epidemiologi Penyakit) Edisi Revisi Tahun 2011, Jakarta: Kemenkes RI, 2012.
- [2] B. RI, “Keamanan Pangan Tanggung Jawab Bersama,” Biro Hukum dan Humas Badan POM RI, Jakarta, 2015.
- [3] Gintoro, I. W. Suharto, F. Rachman, Halim dan Daniel, “Analisis dan Perancangan Sistem Pencarian Taksi Terdekat dengan Pelanggan Menggunakan Layanan Berbasis Lokasi,” dalam *Seminar Nasional Aplikasi Teknologi Informasi 2010 (SNATI 2010)*, Yogyakarta, 2010.
- [4] F. ALJUFRI, Sistem Pemandu Pencarian Masjid Terdekat Berbasis Lokasi di Atas Platform Android, Surabaya, 2012.
- [5] F. P. H. Prasetyo, Pembangunan Aplikasi Pencarian Rumah Sakit Dan Dokter Terdekat Berbasis Android, Yogyakarta: UAJY, 2013.
- [6] S. Kunarti, “Perjanjian Pemborongan Pekerjaan (Outsourcing) dalam Hukum Ketenagakerjaan,” *Jurnal Dinamika Hukum*, vol. 9, 2009.
- [7] I. Zulkarnain, Praktik Outsourcing dan Perlindungan Hak-Hak Pekerja, Simbur Cahaya, Januari 2005.
- [8] R. S. Pressman, Software Engineering, The McGraw Hill, 2005.
- [9] T. P. W. Share, “Food Sharing in the UK,” *The State of the Sharing Economy*, May 2016.
- [10] N. Kishigami, “Contemporary Inuit Food Sharing: A Case Study from Akulivik, PQ. Canada,” 2004.
- [11] T. N. C. Homeless, “Prohibitions on Sharing Food with People Experiencing,” July 2010.
- [12] A. Part, Order To Cash Process Improvement Map, Chicago: Turnaround & Interim Management, 2010.
- [13] S. D. Adeshya, Pembangunan Aplikasi Android untuk Layanan Pesan Antar Barang atau Jasa Berbasis Lokasi dan Teknologi Push Notification, 2012.

- [14] I. G. N. D. Paramartha, "Rancang Bangun Aplikasi Mobile City Directory Yogyakarta Berbasis Android," UAJY, Yogyakarta, 2013.
- [15] C. Ichtiara, "Implementasi Aplikasi Sistem Informasi Geografis (SIG) Universitas Indonesia (UI) Berbasis Web dengan Menggunakan Google Maps API," Depok, Fakultas Teknik Elektro. Universitas Indonesia, 2008.
- [16] A. Eliau, "Layanan Informasi Kereta Api Menggunakan GPS, Google Maps, dan Android," *Jurnal Teknik POMITS No.1*, vol. 01, pp. 1-6, 2012.

BIODATA PENULIS



Penulis lahir di Makassar tanggal 28 Juli 1995. Penulis merupakan anak kedua dari dua bersaudara. Penulis menempuh pendidikan formal di SDN Simomulyo 2 Surabaya, SMP Al-Hikmah Surabaya, dan SMA Muhammadiyah 2 Surabaya. Pada tahun 2013 penulis mengikuti SBMPTN tulis dan diterima di Jurusan Sistem Informasi Institut Teknologi Sepuluh Nopember (ITS)

Surabaya. Sebagai mahasiswa penulis aktif dalam urusan akademik, non akademik maupun organisasi. Tercatat penulis pernah menjadi staff pada Departemen Information Media di Badan Eksekutif Mahasiswa Fakultas Teknologi Sepuluh Nopember dan staff Kementerian Komunikasi dan Informasi di Badan Eksekutif Mahasiswa ITS. Selain itu, penulis juga pernah menjadi Ketua Departemen Kewirausahaan pada Kajian Sistem Informasi (KISI) ITS. Selain organisasi, penulis juga aktif dalam kepanitiaan, baik panitia dalam organisasi maupun di luar organisasi. Penulis juga pernah menjalani kerja praktik di PT. Smart Meter Indonesia di Tangerang selama 1,5 bulan pada tahun 2016.

Pada akhir semester di Jurusan Sistem Informasi, penulis mengambil bidang minat Sistem Enterprise untuk tugas akhir. Untuk kepentingan penelitian, penulis juga dapat dihubungi melalui email faisalsetiap@gmail.com.

Halaman ini sengaja dikosongkan

LAMPIRAN A USABILITY TEST PLAN DASHBOARD

AUTHOR		CONTACT DETAILS		FINAL DATE FOR COMMENTS
FAISAL SETIA PUTRA		faisalsetiap@gmail.com		
PRODUCT UNDER TEST What's being tested? What are the business and experience goals of the product? Website Aplikasi pencariian koki rumah yang terdekat dan tersedia. Aplikasi ini bertujuan untuk menghubungkan antara koki rumah dengan individu.	TEST OBJECTIVES What are the goals of the usability test? What specific questions will be answered? What hypotheses will be tested? Apakah pengguna memahami tujuan dibuatnya aplikasi? Apakah fitur yang disediakan telah sesuai dengan kebutuhan pengguna? Apakah pengguna merasa nyaman dalam menggunakan aplikasi?	PARTICIPANTS How many participants will be recruited? What are their key characteristics? 2 orang yang pernah membantu orang lain memasak dan diberi upah. 2 orang yang paham teknologi dan 2 orang awam. Partisipan terdiri dari laki-laki maupun perempuan.	TEST TASKS What are the test tasks? - Masuk sebagai pengguna - Mendaftarkan diri sebagai koki rumah - Melakukan pemesanan koki rumah - Menambahkan daftar kemampuan memasak - Mengubah daftar kemampuan memasak - Mencari koki berdasarkan kemampuan memasak - Mengubah lokasi tempat tinggal - Melihat riwayat pemesanan jasa - Mencetak tagihan pemesanan jasa - Memberikan rating & komentar kepada koki rumah - Keluar dari aplikasi	RESPONSIBILITIES Who is involved in the test and what are their responsibilities? Faisal Setia Putra (PM, Moderator, client contact)
BUSINESS CASE Why are we doing this test? What are the benefits? What are the risks of not testing? Testing ini bertujuan untuk mendapatkan masukan untuk pengembangan aplikasi selanjutnya	EQUIPMENT What equipment is required? How will you record the data? Laptop dan aplikasi yang terinstall Form responden Akses poin untuk mengakses internet	LOCATION & DATES Where and when will the test take place? When and how will the results be shared? Senin tanggal 02 Januari 2017. Lokasi yang diambil berada di daerah Simo Gunung. Aplikasi dapat diakses di alamat http://localhost/ta		
PROCEDURE What are the main steps in the test procedure?				
<div> <div>Perkenalan identitas diri dan tujuan</div> <div>Pre-test interview</div> <div>Pelaksanaan Test task</div> <div>Pengisian Kuisioner</div> <div>Post-test interview</div> <div>Memberikan ucapan terima kasih dan insentif</div> </div>				

Gambar A-1 Usability Test Plan Dashboard

Halaman ini sengaja dikosongkan

LAMPIRAN B

Hasil Translasi Wawancara

ID	<i>Question and Answer</i>
QA1	<p>Siapa saja yang terlibat dalam pemesanan jasa?</p> <p><i>Bisa individu, organisasi ataupun instansi.</i></p>
QA2	<p>Bagaimana alur proses bisnis pemesanan jasa?</p> <p><i>Selama ini saya dihubungi langsung dengan yang berkepentingan dalam pemesanan jasa.</i></p>
QA3	<p>Siapa saja yang terlibat dalam pembatalan jasa?</p> <p><i>Karena saya ini dipesan langsung sama yang punya kepentingan, jadi jika ada pembatalan, yang berkepentingan langsung menghubungi saya.</i></p>
QA4	<p>Bagaimana alur dari proses pembatalan jasa?</p> <p><i>Biasanya pada proses pembatalan, yang bersangkutan langsung menelpon saya jika membatalkan.</i></p>
QA5	<p>Masalah apa saja yang sering muncul pada proses pemesanan?</p> <p><i>Orang-orang yang butuh jasa saya biasanya agak kesusahan bertemu sama saya karena saya memang jarang di rumah, makanya jika ada pesanan, mereka langsung telpon saya. Selain itu juga kadang biaya jasa yang disepakati di awal tidak dibayar sepenuhnya. Kadang ada juga yang pesan ke saya tempatnya jauh tapi pembayarannya tidak termasuk biaya transport.</i></p>

ID	<i>Question and Answer</i>
QA6	<p>Masalah apa saja yang sering muncul pada proses pembatalan?</p> <p><i>Seringkali pembatalan itu dilakukan secara mendadak, mereka yang membatalkan dengan alasan tertentu tapi ada juga yang mengundurkan jadwalnya.</i></p>
QA7	<p>Apakah terdapat masalah di luar dari proses bisnis?</p> <p><i>Saat ini masih belum ada masalah.</i></p>
QA8	<p>Adakah solusi yang diterapkan untuk menangani masalah pemesanan dan pembatalan tersebut?</p> <p><i>Untuk saat ini saya masih belum bisa menemukan solusi yang tepat untuk menangani masalah pemesanan maupun pembatalan.</i></p>
QA9	<p>Apa harapan terkait solusi atas masalah tersebut?</p> <p><i>Harapan saya ada aplikasi yang dapat manajemen masalah itu semua, baik itu masalah dari pemesanan maupun pembatalan. Jika saya punya aplikasi, pelanggan akan dapat dengan mudah menghubungi orang-orang seperti saya.</i></p>

LAMPIRAN C HASIL PERHITUNGAN SYSTEM USABILITY SCALE (SUS)

Tabel C-1 Hasil Perhitungan SUS

SUS Calculation												
Participant	Jenis Kelamin	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	SUS Score
Ai Juwita	Perempuan	4	1	5	2	5	2	4	1	4	1	87,5
Vallian Owen	Laki-Laki	4	2	5	2	4	1	4	1	5	2	85
Nadya Chandra	Perempuan	4	1	5	1	4	2	4	1	4	1	87,5
Eis Daryati	Perempuan	3	1	4	2	5	1	4	1	4	2	82,5
Winarto	Laki-Laki	4	1	5	1	5	2	5	1	5	1	95
Rahmawati	Perempuan	4	1	5	1	5	1	4	1	4	2	90
Total Usability												87,92

Halaman ini sengaja dikosongkan

LAMPIRAN D KODE PEMBUATAN APLIKASI

```
<?php
include 'connect.php';
require_once "mail/PHPMailerAutoload.php";

$id_pesan = $_POST['id_pesan'];
$id_user = $_POST['id_user'];
$id_koki = $_POST['id_koki'];

$sql_jasa = "SELECT * FROM 'pemesanan' WHERE id_koki='$id_koki' AND id_customer='$id_user' AND is_booking='1'";
$result_jasa = $conn->query($sql_jasa);

if(isset($_POST['submit'])) {
    $email_koki = $_POST['email_koki'];
    $subtotal = $_POST['subtotal'];
    $total = $_POST['total'];

    if(mysqli_num_rows($result_jasa)>0) {
        echo "
        <script>
            alert('Maaf pemesanan telah dilakukan');
        </script>";
    }
    else {
        $sql = "UPDATE 'pemesanan' SET 'is_verify'='1', 'is_booking'='1' WHERE 'id_pemesanan' = '$id_pesan'";
        $result = $conn->query($sql);

        $sql_mail = "SELECT id_user, nama_lengkap, email FROM 'user' WHERE id_user='$id_user'";
        $result_mail = $conn->query($sql_mail);
        $row = $result_mail->fetch_assoc();

        $to = $row['email'];
        $from = $email_koki;
        $first_name = $row['nama_lengkap'];
```

Gambar D-1 Verifikasi Pemesanan

```
<?php
include 'connect.php';

$id_user = $_POST['id_user'];
$skill = $_POST['skill'];
$deskripsi = $_POST['deskripsi'];
$price = $_POST['price'];

$sql = "UPDATE 'jasa' SET 'daftar_jasa'='$skill', 'price'='$price', 'deskripsi_jasa'='$deskripsi' WHERE id_koki='$id_user'";
$result = $conn->query($sql);

if(isset($_POST['submit'])) {
    if($result) {
        if (isset($_POST['available'])) {
            $sql = "UPDATE 'user' SET 'is_koki'='1' WHERE id_user='$id_user'";
            $result = $conn->query($sql);
        }
        else {
            $sql = "UPDATE 'user' SET 'is_koki'='0' WHERE id_user='$id_user'";
            $result = $conn->query($sql);
        }
        echo "
        <script>
            alert('Informasi jasa berhasil diubah');
            window.history.back();
        </script>";
    }
    else {
        echo "
        <script>
            alert('Maaf informasi jasa gagal diubah');
            window.history.back();
        </script>";
    }
}
```

Gambar D-2 Mengubah Kemampuan Koki Rumahan

```

<?php
include 'connect.php';

$id_user = $_POST['id_user'];
$fullname = $_POST['fullname'];
$tanggal = $_POST['tanggal_lahir'];
$gender = $_POST['jenis_kelamin'];
$alamat = $_POST['alamat'];
$phone = $_POST['phone'];
$lat = $_POST['latitude'];
$long = $_POST['longitude'];

$sql = "UPDATE `user` SET `nama_lengkap`='$fullname', `tanggal_lahir`='$tanggal', `jenis_kelamin`='$gender'
$result = $conn->query($sql);

$target_dir = "uploads/";
$target_file = $target_dir . basename($_FILES["user_photo"]["name"]);
$target_name = $target_dir . "user-$id_user.jpg";
$uploadOk = 1;
$imageFileType = pathinfo($target_file,PATHINFO_EXTENSION);

if(isset($_POST['submit'])){
    $check = getimagesize($_FILES["user_photo"]["tmp_name"]);
    if($check !== false) {
        echo "File is an image - " . $check["mime"] . ".";
        $uploadOk = 1;
    } else {
        echo "File is not an image.";
        $uploadOk = 0;
    }
    if($result)
    {
        echo "
        <script>
        alert('Informasi pribadi berhasil diubah');
        window.history.back();

```

Gambar D-3 Mengubah Informasi Pribadi

```

<?php
include('header.php');
include('header_start.php');

$sql_check = "SELECT * FROM `user` WHERE id_user='$id_user' AND role='2'";
$result_check = $conn->query($sql_check);
if(mysqli_num_rows($result_check)){
    $sql = "SELECT * FROM `user` AS u, `jasa` AS j WHERE u.id_user='$id_user' AND j.id_koki='$id_user'";
} else {
    $sql = "SELECT * FROM `user` WHERE id_user='$id_user'";
}
$result = $conn->query($sql);
$row = $result->fetch_assoc();
?>

<!-- ##### PAGE START-->
<div class="row-col">
    <div class="col-sm-3 col-lg-2 b-r">
        <div class="nav-active-border left b-primary">
            <ul class="nav nav-sm">
                <li class="nav-item">
                    <a class="nav-link block active" href="#" data-toggle="tab" data-target="#tab-1">Informasi Diri</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link block" href="#" data-toggle="tab" data-target="#tab-2">Keamanan</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link block" href="#" data-toggle="tab" data-target="#tab-3">Koki</a>
                </li>
            </ul>
        </div>
    </div>
</div>

```

Gambar D-4 Mengubah Pengaturan Informasi


```

<?php include('header.php'); ?>

<div class="padding">
  <div class="navbar">
    <div class="pull-center">
      <!-- brand -->
      <a href="index" class="navbar-brand">
        <div data-ui-include="images/logo.svg"></div>
        
        <span class="hidden-folded inline">Mama Cooks</span>
      </a>
      <!-- / brand -->
    </div>
  </div>
</div>
<div class="b-t">
  <div class="center-block w-xxl w-auto-xs p-y-md text-center">
    <div class="p-a-md">

      <form name="form" action="create_account" method="post">
        <div class="form-group">
          <input type="text" name="fullname" class="form-control" placeholder="Nama Lengkap" required>
        </div>
        <div class="form-group">
          <input type="email" name="email" class="form-control" placeholder="Email" required>
        </div>
        <div class="form-group">
          <input type="password" name="password" class="form-control" placeholder="Kata Sandi" required>
        </div>
        <div class="form-group">
          <label class="md-check p-x-sm">
            <input type="radio" name="jenis_kelamin" value="Laki-Laki" checked>
            <i class="blue"></i>
            Laki-Laki
          </label>
          <label class="md-check">
            <input type="radio" name="jenis_kelamin" value="Perempuan">
            <i class="blue"></i>
            Perempuan
          </label>
        </div>
      </form>
    </div>
  </div>
</div>

```

Gambar D-5 Form Pendaftaran Pelanggan

```

<form name="form" action="login" method="post">
  <div class="form-group">
    <input type="email" name="email" class="form-control" placeholder="Email" required>
  </div>
  <div class="form-group">
    <input type="password" name="password" class="form-control" placeholder="Kata Sandi" required>
  </div>
  <div class="m-b-md">
    <label class="md-check">
      <input type="checkbox"><i class="primary"></i> Keep me signed in
    </label>
  </div>
  <button type="submit" name="submit" class="btn btn-lg black p-x-lg">Masuk ke Mama Cooks</button>
</form>
<div class="m-y">
  <a href="forgot-password" class="_600">Lupa password Anda?</a>
</div>
<div>
  Belum punya akun Mama Cooks? Daftar
  <a href="signup" class="text-primary _600">di sini</a>
</div>
:/div>

```

Gambar D-6 Form Login Aplikasi

```
<?php
include 'connect.php';

if(isset($_POST['masakan'])){
    $masakan = $_POST['masakan'];
    header("Location: home?masakan=$masakan");
}
?>
```

Gambar D-7 Pencarian Koki Berdasarkan Kemampuan

```
$id_user = $_POST['id_user'];
$skill = $_POST['skill'];
$deskripsi = $_POST['deskripsi'];
$price = $_POST['price'];

if(isset($_POST['submit'])) {
    if(isset($_POST['check'])) {
        $sql = "INSERT INTO `jasa` (`id_koki`, `daftar_jasa`, `price`, `deskripsi_jasa`) VALUES ('$id_user', '$skill', '$price', '$deskripsi')";
        $result = $conn->query($sql);
        if($result) {
            $sql_update = "UPDATE `user` SET `is_koki`='1', `role`='2' WHERE `id_user`='$_POST[id_user]'";
            $result_update = $conn->query($sql_update);
            echo "
            <script>
                alert('Selamat, Anda berhasil terdaftar menjadi koki rumahan');
                window.location.href = 'home/';
            </script>";
        }
        else {
            echo "
            <script>
                alert('Maaf data gagal disimpan');
                window.history.back();
            </script>";
        }
    }
    else {
        echo "
        <script>
            alert('Maaf klik setuju terlebih dahulu');
            window.history.back();
        </script>";
    }
}
```

Gambar D-8 Registrasi Koki Rumahan

```

<?php
include('header.php');
include('header_start.php');
if(isset($_GET['rate'])){
    $id_pemesanan = $_GET['rate'];
}

<!-- ##### PAGE START-->
<div class="padding">

    <div class="row">
        <div class="col-md-6 offset-md-3">
            <form action="koki_rating" method="post" enctype="multipart/form-data" data-ui-jp="parsley">
                <div class="box">
                    <input type="hidden" name="id_pemesanan" value="php echo $id_pemesanan; ?">
                    <div class="box-body p-t-lg">
                        <div class="form-group text-center">
                            <h3 class="font-weight-normal">Realitas dan Rasa Makanan</h3>
                            <div class="stars">
                                <input type="radio" name="star" class="star-1" onclick="postToController()" value="1" id="star-1" />
                                <input type="radio" name="star" class="star-2" onclick="postToController()" value="2" id="star-2" />
                                <input type="radio" name="star" class="star-3" onclick="postToController()" value="3" id="star-3" />
                                <input type="radio" name="star" class="star-4" onclick="postToController()" value="4" id="star-4" />
                                <input type="radio" name="star" class="star-5" onclick="postToController()" value="5" id="star-5" />
                            </div>
                        </div>
                        <div class="form-group">
                            <label>Kritik dan Saran</label>
                            <textarea class="form-control" name="komentar" rows="6" data-minwords="6" placeholder="Rasa makanannya terlalu hambar">
                        </div>
                    </div>
                </div>
            </form>
        </div>
    </div>

```

Gambar D-9 Form Pemberian Rating dan Komentar

```

<?php
include('header.php');
include('header_start.php');

if(isset($_GET['jasa'])){
    $id_jasa = $_GET['jasa'];
    $id_koki = $_GET['user'];
    $distance = $_GET['d'];

    $sql_check = "SELECT * FROM 'pemesanan' WHERE id_jasa='$id_jasa' AND is_verify='0' OR is_verify='1'";
    $result_check = $conn->query($sql_check);
    if(mysqli_num_rows($result_check)){
        $sql_jasa = "SELECT * FROM 'jasa' AS j, 'user' AS u WHERE j.id_jasa='$id_jasa' AND u.id_user='$id_koki'";
        $result_jasa = $conn->query($sql_jasa);
        $row_jasa = $result_jasa->fetch_assoc();

        $sharga = $row_jasa['price'];
        $id_koki = $row_jasa['id_koki'];
    }
    else {
        $sql_jasa = "SELECT * FROM 'jasa' AS j, 'user' AS u WHERE j.id_jasa='$id_jasa' AND u.id_user='$id_koki'";
        $result_jasa = $conn->query($sql_jasa);
        $row_jasa = $result_jasa->fetch_assoc();

        $sharga = $row_jasa['price'];
        $id_koki = $row_jasa['id_koki'];

        $sql = "INSERT INTO 'pemesanan' ('id_koki', 'id_customer', 'id_jasa', 'transport_price', 'total_price', 'is_verify'
        $result = $conn->query($sql);
    }

    $sql_koki = "SELECT id_user, nama_lengkap, email FROM 'user' WHERE id_user='$id_koki'";
    $result_koki = $conn->query($sql_koki);
    $row_koki = $result_koki->fetch_assoc();
}

```

Gambar D-10 Verifikasi Awal Pemesanan

```

<form action="verify_order" method="post">
<table class="table table-striped white b-a">
  <thead>
    <tr>
      <th>DESKRIPSI</th>
      <th style="width: 150px">HARGA JASA</th>
      <th style="width: 150px">TOTAL</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>
        <input type="hidden" name="id_pesan" value="{?php echo $row_select['id_pemesanan']; }?">
        <input type="hidden" name="id_koki" value="{?php echo $row_koki['id_koki']; }?">
        <input type="hidden" name="id_user" value="{?php echo $id_user; }?">
        <td>Jasa memasak makanan a/n {?php echo $row_koki['nama_lengkap']; }?</td>
        <td>Rp {?php echo number_format($harga,0,',','.'); }?</td>
        <td>Rp {?php echo number_format($harga,0,',','.'); }?</td>
      </tr>
      <tr>
        <td colspan="2" class="text-right"><strong>Subtotal</strong></td>
        <td>Rp {?php echo number_format($harga,0,',','.'); }?</td>
      </tr>
      <tr>
        <td colspan="2" class="text-right no-border"><strong>Transport</strong></td>
        <td>Rp {?php echo number_format(($distance*2500),0,',','.'); }?</td>
      </tr>
      <tr>
        <td colspan="2" class="text-right no-border"><strong>Total</strong></td>
        <td><strong>Rp {?php echo number_format($harga+($distance*2500),0,',','.'); }?</strong></td>
      </tr>
    </tbody>
  </table>
  <div>
    <button type="submit" name="delete" class="btn"><i class="fa fa-times"></i> Batalkan</button>
    <button type="button" name="submit" data-toggle="modal" data-target="#m" class="btn info pull-right">
  </div>
</form>

```

Gambar D-11 Form Verifikasi Awal Pemesanan

```

<div class="row-col">
  <div class="col-lg b-r">
    <div class="padding">
      <?php
        $result = $conn->query($sql);
        if ( $result && mysqli_num_rows($result)>0 ) {
          while ( $row = $result->fetch_assoc() ) {
            $customer = $row['id_customer'];
            $koki = $row['id_koki'];
            if ($koki==$id_user) {
              $sql_cust = "SELECT * FROM 'user' WHERE 'id_user'='$customer'";
            } else {
              $sql_cust = "SELECT * FROM 'user' WHERE 'id_user'='$koki'";
            }
            $result_cust = $conn->query($sql_cust);
            $row_cust = $result_cust->fetch_assoc();
            if ($customer==$id_user && $row['is_booking']=='1' && $row['is_finish']=='0') {
              ?>
            }
          }
        }
      </?php>
      <form class="" method="post" class="row-col b-a white m-b">
        <div class="col-md-2 b-r no-border-sm">
          <div class="text-center">
            <div class="p-a-md">
              <p></p>
              <a href="#" class="text-md block"><?php echo $row_cust['nama_lengkap']; }?</a>
              <p><small><?php echo $row_cust['alamat']; }?</small></p>
              <p><small><?php echo $row_cust['phone']; }?</p>
            </div>
            <div class="p-b-md">
              <?php if ($row['is_approve']=='0') {?>
              <span class="label label-lg warning">Menunggu Persetujuan</span>
              <?php } else {?>
              <a href="rating?rate={?php echo $row['id_pemesanan']; }?" class="btn info">Verifikasi Selesai</a>
              <?php } ?>
            </div>
          </div>
        </div>
      </form>
    </div>
  </div>

```

Gambar D-12 Notifikasi Pemesanan

```

if(isset($_POST['submit'])) {
    $id_pemesanan = $_POST['id_pemesanan'];
    $sql = "SELECT * FROM `pemesanan` WHERE `id_pemesanan` = '$id_pemesanan'";
    $result = $conn->query($sql);
    $row = $result->fetch_assoc();

    $id_koki = $row['id_koki'];
    $id_customer = $row['id_customer'];
    $id_jasa = $row['id_jasa'];
    $star = $_POST['star'];
    $komentar = $_POST['komentar'];

    $sql_insert = "INSERT INTO `feedback` (`id_pemesanan`, `id_koki`, `id_customer`, `id_jasa`, `count_star`, `komentar`) VALUES
    ($result_insert = $conn->query($sql_insert);
    if($result_insert) {
        $sql_insert = "UPDATE `pemesanan` SET `is_booking` = '0', `is_finish` = '1' WHERE `id_pemesanan` = '$id_pemesanan'";
        $result_insert = $conn->query($sql_insert);
        echo "
            <script>
                alert('Terima kasih atas masukan Anda');
                window.location.href = 'home';
            </script>";
    }
    else {
        echo "
            <script>
                alert('Maaf gagal memberikan masukan');
                window.history.back();
            </script>";
    }
}

```

Gambar D-13 Rating Koki

```

<?php
include('header.php');
include('header_start.php');
if(!isset($_GET['inv'])) {
    echo "<script>window.history.back();</script>";
}
else {
    $id = $_GET['inv'];
    $sql = "SELECT * FROM `pemesanan` WHERE `id_pemesanan` = '$id'";
    $result = $conn->query($sql);
    $row = $result->fetch_assoc();
    ?>

<!-- ##### PAGE START-->
<div class="padding">
    <a href="#" class="btn btn-sm btn-info pull-right hidden-print" onClick="window.print();">Cetak</a>
    <p></p>
    <div class="row">
        <div class="col-xs-6">
            <h4 class="text-md">Mama Cooks.</h4>
            <p>Simo Gunung Barat Tol <br>
                60181 Surabaya, Jawa Timur<br>
                Indonesia
            </p>
            <p>Email: admin@mamacooks.com</p>
        </div>
    </div>
    ...

```

Gambar D-14 Form Tagihan

```

<?php
if(mysqli_num_rows($result) > 0){
while($rows = $result->fetch_assoc()){
    $koki = $rows['id_koki'];
    $detail = "SELECT * FROM `user` WHERE id_user='$koki'";
    $result_detail = $conn->query($detail);
    $row = $result_detail->fetch_assoc();
    if($rows['is_finish']=='1'){?>
    <div class="sl-item b-success">
    <div class="sl-icon b-success">
    <i class="fa fa-check"></i>
    </div>
    <div class="sl-content">
    <div class="sl-date text-muted"><?php echo date("j F Y", strtotime($rows['booking_time'])); ?>
    <p class="m-b-xs"><a href="invoice?inv=<?php echo $rows['id_pemesanan']; ?>"><?php echo "#INV"
    <span class="label label-md success">Lunas</span>
    </div>
    </div>
<?php } else if($rows['is_verify']=='1' && $rows['is_booking']=='1' && $rows['is_approve']=='0') {
?>
<div class="sl-item b-warning">
<div class="sl-icon b-warning">
<i class="fa fa-refresh"></i>
</div>
<div class="sl-content">
<div class="sl-date text-muted"><?php echo date("j F Y", strtotime($rows['booking_time'])); ?>
<p class="m-b-xs"><a href="invoice?inv=<?php echo $rows['id_pemesanan']; ?>"><?php echo "#INV"
<span class="label label-sm warning">Proses</span>
</div>
</div>
<?php } else {

```

Gambar D-15 Riwayat Pemesanan

```

$fullname = $_POST['fullname'];
$email = $_POST['email'];
$gender = $_POST['jenis_kelamin'];
$password = sha1(mysqli_real_escape_string($conn,($_POST['password'])));

$sql = "SELECT email FROM `user` WHERE `email`='$_email'";
$result = $conn->query($sql);

if(isset($_POST['submit'])){
    if(mysqli_num_rows($result)>0){
        echo "
        <script>
        alert('Maaf Email telah terdaftar');
        window.history.back();
        </script>";
    }
    else {
        $sql = "INSERT INTO `user` ('nama_lengkap', 'email', 'jenis_kelamin', 'password', 'role') VALUES
        $result = $conn->query($sql);
        if($result)
        {
            echo "
            <script>
            alert('Selamat, Anda telah terdaftar');
            window.location.href = 'signin';
            </script>";
        }
        else
        {
            echo "
            <script>
            alert('Maaf Akun gagal terdaftar');
            window.history.back();
            </script>";
        }
    }
}

```

Gambar D-16 Simpan Akun

LAMPIRAN E LEMBAR PERSETUJUAN

Pendahuluan

Testing ini dilakukan untuk melakukan penilaian terhadap kegunaan (Usabilitas) dari aplikasi yang akan dibuat oleh peneliti.

Aplikasi yang akan dibuat diberi nama Mama Cooks. Mama Cooks merupakan aplikasi pencarian koki rumahan yang terdekat dan tersedia.

Pengembangan aplikasi Mama Cooks memiliki tujuan yaitu untuk menghubungkan antara koki rumahan dengan individu.

Usability testing ini akan memakan waktu sekitar 20 menit dan media yang digunakan adalah aplikasi *Minimum Viable Product*. Harapannya hasil masukan dari testing akan dibuat untuk mengembangkan aplikasi kedepannya. seluruh fitur dan fungsi dari aplikasi dibuat berdasarkan literatur dan wawancara dengan pihak yang terkait dengan penelitian ini.

Terdapat 2 Bagian pertanyaan yang harus dijawab. Pada bagian pertama menggunakan tipe soal likert dengan skala 1 – 5 yang bertujuan mengukur usabilitas dari aplikasi. Pada bagian kedua menggunakan tipe soal bebas, untuk mengambil evaluasi dan saran untuk pengembangan aplikasi.

Terima kasih atas partisipasi anda.

FORM PERSETUJUAN

Saya mengerti bahwa partisipasi dalam studi kegunaan ini bersifat sukarela.

Silakan isi form berikut untuk menunjukkan bahwa Anda telah membaca dan Anda memahami informasi pada formulir ini dan bahwa setiap pertanyaan yang mungkin Anda miliki tentang sesi telah dijawab.

Tanggal : _____

Nama Lengkap : _____

Jenis Kelamin : _____

Usia : _____

Tanda tangan : _____

LAMPIRAN F SYSTEM USABILITY SCALE FORM

Instruksi : Isikan setiap jawaban pernyataan dengan memberikan tanda *checklist* (√) atau *cross* (x) pada salah satu kolom 1 – 5 yang menurut Anda paling sesuai dengan pendapat Anda.

1 = Sangat tidak setuju

5 = Sangat setuju

Pernyataan	1	2	3	4	5
Saya berpikir bahwa saya ingin menggunakan sistem ini sesering mungkin					
Saya berpikir sistem ini sangat rumit dan tidak perlu					
Saya berpikir sistem ini mudah digunakan					
Saya berpikir saya akan membutuhkan bantuan seseorang untuk dapat menggunakan sistem ini					
Saya menemukan seluruh fungsi yang telah terintegrasi dengan sangat baik					
Saya berpikir terlalu banyak inkonsistensi dari sistem ini					
Saya berpikir banyak orang yang akan dapat belajar menggunakan sistem ini dengan cepat					
Saya berpikir bahwa sistem ini sangat sulit untuk digunakan					
Saya merasa sangat percaya diri dalam menggunakan sistem ini					
Saya perlu belajar banyak hal sebelum dapat menggunakan sistem ini					

BAGIAN II EVALUASI

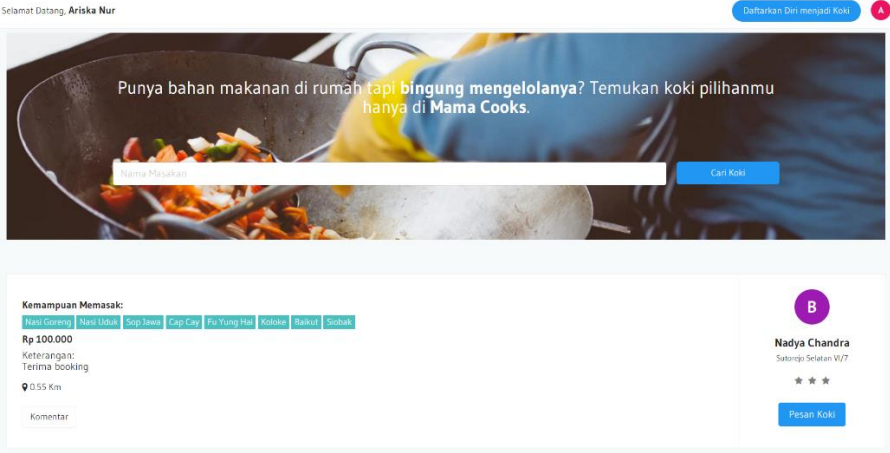
1. Adakah fungsi dari aplikasi yang anda rasa tidak perlu ada? sebutkan

2. Adakah fungsi dari sistem yang menurut Anda penting namun belum terdapat pada aplikasi? Sebutkan

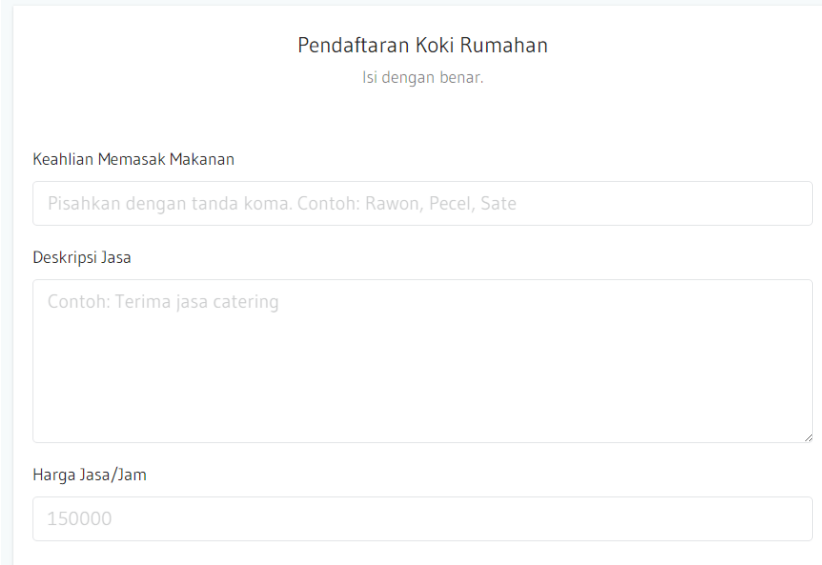
3. Saran anda untuk perbaikan kedepannya?

LAMPIRAN G SKENARIO TEST CASE

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC01	Login	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Masuk sebagai pengguna			
Step	1. Memasukkan email pada kolom email. 2. Memasukkan password pada kolom password 3. Tekan Tombol Masuk.			
Input	Email dan Password			
Hasil Ekspektasi	1. Sistem menampilkan halaman daftar koki rumahan terdekat.			

<p>Hasil Testing</p>	<p>1. Sistem menampilkan halaman awal daftar koki rumahan terdekat.</p>
<p>Bukti</p>	 <p>The screenshot shows the home screen of the 'Mama Cooks' application. At the top, there is a greeting 'Selamat Datang, Ariska Nur' and a button to 'Daftarkan Diri menjadi Koki'. Below this is a large banner with the text 'Punya bahan makanan di rumah tapi bingung mengelolanya? Temukan koki pilihanmu hanya di Mama Cooks.' and a 'Cari Koki' button. Under the banner, there is a section titled 'Kemampuan Memasak:' with a list of cuisines: Nasi Goreng, Nasi Uduk, Sop Jawa, Capr Cay, Fu Yang Mai, Kotoku, Bakut, and Sotok. The price is listed as 'Rp 100.000' and the location as '0.55 Km'. On the right, there is a profile card for 'Nadya Chandra' with a rating of 3 stars and a 'Pesan Koki' button.</p>

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC02	Koki Rumahan dapat mendaftarkan diri	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Mendaftarkan diri sebagai koki rumahan			
Step	<div>1. Menekan tombol Daftarkan diri sebagai koki rumahan</div> <div>2. Memasukkan keahlian memasak, deskripsi jasa dan harga jasa per jam</div> <div>3. Menekan tanda setuju terhadap kebijakan layanan</div> <div>4. Menekan tombol Submit</div>			
Input	Keahlian memasak, deskripsi jasa, harga jasa per jam			
Hasil Ekspektasi	<div>1. Sistem menampilkan halaman form pendaftaran koki.</div> <div>2. Sistem menampilkan pesan berhasil resgistrasi menjadi koki rumahan</div> <div>3. Sistem mengarahkan ke halaman home</div>			
Hasil Testing	<div>1. Sistem menampilkan halaman form pendaftaran koki.</div>			

	<ol style="list-style-type: none">2. Sistem menampilkan pesan berhasil registrasi menjadi koki rumahan3. Sistem mengarahkan ke halaman home
Bukti	

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC03	Pelanggan dapat memesan jasa koki rumahan	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Melakukan pemesanan koki rumahan			
Step	<div>1. Memilih koki yang sesuai dengan kebutuhan</div> <div>2. Menekan tombol Pesan Koki</div> <div>3. Menambahkan informasi tambahan jika diperlukan</div> <div>4. Menekan tombol Lanjutkan Pemesanan</div> <div>5. Menekan tombol konfirmasi Pesan</div>			
Input	Jasa Koki Rumahan			
Hasil Ekspektasi	<div>1. Sistem menampilkan halaman daftar koki.</div> <div>2. Sistem menampilkan halaman detail pemesanan.</div> <div>3. Sistem menampilkan dialog konfirmasi pemesanan</div> <div>4. Sistem mengirimkan email kepada pemesan dan koki rumahan.</div>			

Hasil Testing

1. Sistem menampilkan halaman daftar koki.
2. Sistem menampilkan halaman detail pemesanan.
3. Sistem menampilkan dialog konfirmasi pemesanan
4. Sistem mengirimkan email kepada pemesan dan koki rumahan.

Bukti

DESKRIPSI	HARGA JASA	TOTAL
Jasa memasak makanan a/n Faisal Setia Putra	Rp 85.000	Rp 85.000
	Subtotal	Rp 85.000
	Transport	Rp 12.500
	Total	Rp 97.500

Form Salinan Tagihan



Inbox x

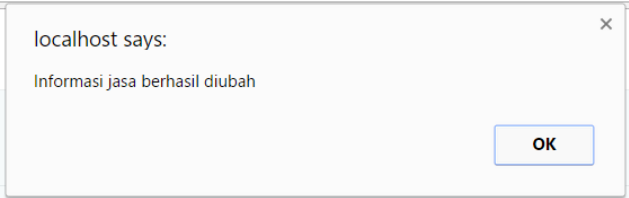


Mama Cooks <faisal.isol@gmail.com>

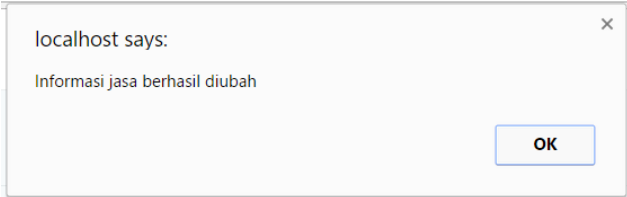
to me ▾

Ariska Nur Memesan jasa anda dengan: Subtotal: **85.000**, Total Harga: **97.500**
Informasi Tambahan:

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC04	Koki rumahan dapat memasukkan kemampuan memasak jenis makanan	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Menambahkan daftar kemampuan memasak			
Step	<div>1. Menekan menu pengaturan</div> <div>2. Memilih tab Koki</div> <div>3. Menambahkan Keahlian memasak pada isian keahlian memasak makanan</div> <div>4. Menekan tombol Simpan Perubahan</div>			
Input	Data koki dan keahliannya			
Hasil Ekspektasi	<div>1. Sistem menampilkan halaman pengaturan koki.</div> <div>2. Sistem menampilkan informasi pengubahan informasi berhasil.</div> <div>3. Sistem mengarahkan ke halaman pengaturan</div>			

Hasil Testing	<ol style="list-style-type: none">1. Sistem menampilkan halaman pengaturan koki.2. Sistem menampilkan informasi pengubahan jasa berhasil.3. Sistem mengarahkan ke halaman pengaturan
Bukti	

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC05	Koki rumahan dapat memasukkan kemampuan memasak jenis makanan	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Mengubah daftar kemampuan memasak			
Step	1. Menekan menu pengaturan 2. Memilih tab Koki 3. Mengubah Keahlian memasak pada isian keahlian memasak makanan 4. Menekan tombol Simpan Perubahan			
Input	Data koki dan keahliannya			
Hasil Ekspektasi	1. Sistem menampilkan halaman pengaturan koki. 2. Sistem menampilkan informasi pengubahan informasi berhasil. 3. Sistem mengarahkan ke halaman pengaturan			

Hasil Testing	<ol style="list-style-type: none">1. Sistem menampilkan halaman pengaturan koki.2. Sistem menampilkan informasi pengubahan jasa berhasil.3. Sistem mengarahkan ke halaman pengaturan
Bukti	

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC06	Koki rumahan dapat memasukkan kemampuan memasak jenis makanan	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Mencari koki berdasarkan kemampuan memasak			
Step	<div>1. Mencari koki sesuai dengan kebutuhan</div> <div>2. Memasukkan kemampuan koki pada isian pencarian di halaman awal</div> <div>3. Menekan tombol Cari Koki</div>			
Input	Data koki dan keahliannya			
Hasil Ekspektasi	<div>1. Sistem menampilkan halaman daftar koki</div> <div>2. Sistem menampilkan halaman daftar koki sesuai dengan inputan kemampuan koki yang dicari</div>			

Hasil Testing

1. Sistem menampilkan halaman daftar koki
2. Sistem menampilkan halaman daftar koki sesuai dengan inputan kemampuan koki yang dicari

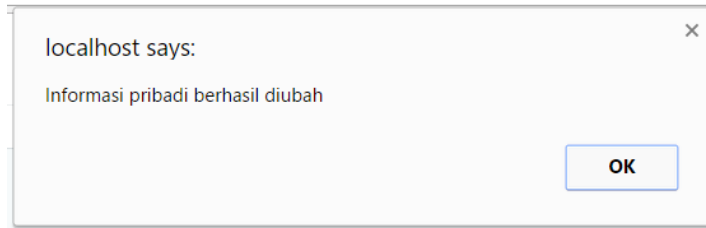
Bukti



<p>Kemampuan Memasak:</p> <p>Nasi Goreng Nasi Uduk Sop Iken Cap Cay Paku Tenggir Rendang Bakso Soto</p> <p>Rp.100.000</p> <p>Keterangan: Terima booking</p> <p>📍 0.55 Km</p> <p>Komentar</p>	<p>B</p> <p>Nadya Chandra Suburmu Setoran 10/7</p> <p>★ ★ ★</p> <p>Pesan Koki</p>
<p>Kemampuan Memasak:</p> <p>Soto Gado Soto Nasi Goreng</p> <p>Rp.100.000</p> <p>Keterangan: Terima pesanan makanan</p> <p>📍 1.43 Km</p> <p>Komentar</p>	<p>B</p> <p>mfahmizamroni Jl. Bujangan</p> <p>★</p> <p>Pesan Koki</p>

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC07	Koki rumahan dapat mengubah koordinat lokasi tempat tinggal	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Mengubah Lokasi Tempat Tinggal			
Step	1. Masuk ke menu pengaturan 2. Memasukkan lokasi pada inputan Lokasi 3. Menekan tombol Simpan Perubahan			
Input	Data koki dan keahliannya			
Hasil Ekspektasi	1. Sistem menampilkan halaman pengaturan 2. Sistem menampilkan informasi perubahan informasi pribadi berhasil			
Hasil Testing	1. Sistem menampilkan halaman pengaturan 2. Sistem menampilkan informasi perubahan informasi pribadi berhasil			

Bukti



Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC08	Koki rumahan mendapatkan daftar riwayat pemesanan jasa	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Melihat riwayat pemesanan jasa			
Step	1. Masuk ke menu history 2. Menekan pilihan transaksi jasa yang pernah dipesan			
Input	Data koki dan keahliannya			
Hasil Ekspektasi	1. Sistem menampilkan halaman riwayat pemesanan 2. Sistem menampilkan halaman invoice			
Hasil Testing	1. Sistem menampilkan halaman riwayat pemesanan 2. Sistem menampilkan halaman invoice			

Bukti


Riwayat Pemesanan

- 7 January 2017
#INV11 Pemesanan a/n Nadya Chandra
Lunas
- 9 January 2017
#INV70 Pemesanan a/n Faisal Setia Putra
Ditolak
- 10 January 2017
#INV71 Pemesanan a/n Faisal Setia Putra
Lunas
- 9 January 2017
#INV72 Pemesanan a/n Faisal Setia Putra
Ditolak
- 9 January 2017
#INV73 Pemesanan a/n Faisal Setia Putra
Ditolak
- 11 January 2017
#INV74 Pemesanan a/n Faisal Setia Putra
Proses

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC09	Koki rumahan mendapatkan informasi tagihan pelanggan	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Mencetak tagihan pemesanan jasa			
Step	1. Masuk ke menu history 2. Menekan pilihan transaksi jasa yang pernah dipesan 3. Menekan tombol Cetak			
Input	Data koki dan keahliannya			
Hasil Ekspektasi	1. Sistem menampilkan halaman riwayat pemesanan 2. Sistem menampilkan halaman invoice 3. Sistem menampilkan preview cetak			
Hasil Testing	1. Sistem menampilkan halaman riwayat pemesanan			

2. Sistem menampilkan halaman invoice
3. Sistem menampilkan preview cetak

Bukti



Mama Cooks.
 Simo Gunung Barat Tol
 60181 Surabaya, Jawa Timur
 Indonesia Indonesia
 Email: admin@mamacooks.com

Cetak

#INV71

10 January 2017

Lunas

CUSTOMER:

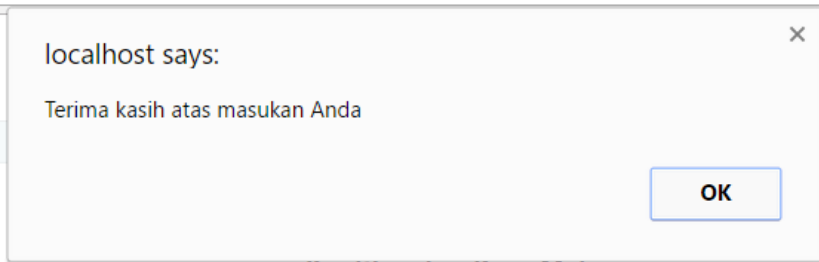
rahma
simo gunung barat

KOKO RUMAHAN:

Faisal Setia Putra
Simo Gunung Barat Tol 2/11

DESKRIPSI	HARGA JASA	TOTAL
Jasa memasak makanan a/n Faisal Setia Putra	Rp 85.000	Rp 85.000
	Subtotal	Rp 85.000
	Transportasi	Rp 10.000
	Total	Rp 95.000

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC10	Pelanggan dapat memberikan rating dan komentar kepada koki rumahan	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Memberikan rating dan komentar kepada koki rumahan			
Step	<div>1. Transaksi telah selesai</div> <div>2. Menekan tombol Verifikasi Selesai</div> <div>3. Mengisi kritik dan saran serta memberikan rating</div> <div>4. Menekan tombol Rate</div>			
Input	Data koki dan keahliannya			
Hasil Ekspektasi	<div>1. Sistem menampilkan halaman rating dan komentar</div> <div>2. Sistem menampilkan informasi pesan terima kasih</div> <div>3. Sistem menampilkan halaman daftar koki rumahan</div>			

Hasil Testing	<ol style="list-style-type: none">1. Sistem menampilkan halaman rating dan komentar2. Sistem menampilkan informasi pesan terima kasih3. Sistem menampilkan halaman daftar koki rumahan
Bukti	 <p>The screenshot shows a standard JavaScript alert dialog box. The title bar reads 'localhost says:'. The main text area contains the message 'Terima kasih atas masukan Anda' in a blue font. At the bottom right, there is a single button labeled 'OK'.</p>

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
TC11	Logout	Sukses	02 Januari 2017	Partisipan
			12.00 WIB	
Objective	Keluar dari aplikasi			
Step	1. Memilih menu keluar			
Input	Data koki dan keahliannya			
Hasil Ekspektasi	1. Sistem menampilkan halaman home			
Hasil Testing	1. Sistem menampilkan halaman home			

Bukti

